AEC RESEARCH AND DEVELOPMENT REPORT

Y-1660
Mathematics
and Computers

GEODESIC PATHS ON SURFACES OF REVOLUTION: A COMPUTER-AIDED FILAMENT-WINDING DESIGN PROGRAM

(Prepared for Sandia Corporation under Purchase Order ASB-92-1849)

T. W. Bookhart
A. H. Fowler

UNION CARBIDE CORPORATION
NUCLEAR DIVISION
OAK RIDGE Y-12 PLANT

operated for the ATOMIC ENERGY COMMISSION under U. S. GOVERNMENT Contract W-7405

UNION CARBIDE
OAK RIDGE Y-12 PLANT
P. O. Box Y
OAK RIDGE, TENNESSEE 37830

Printed in the United States of America. Available from Clearinghouse for Federal
Scientific and Technical Information, National Bureau of Standards,
U.S. Department of Commerce, Springfield, Virginia 22151
Price: Printed Copy $3.00; Microfiche $0.65

Date Issued: November 20, 1968

Document Y-1660

Mathematics and Computers
TID-4500

UNION CARBIDE CORPORATION

Nuclear Division

Y-12 PLANT

Contract W-7405-eng-26
With the US Atomic Energy Commission

GEODESIC PATHS ON SURFACES OF REVOLUTION: A COMPUTER-AIDED FILAMENT-WINDING DESIGN PROGRAM

(Prepared for Sandia Corporation under Purchase Order ASB-92-1849)

T. W. Bookhart
A. H. Fowler

Oak Ridge, Tennessee

Document Y-1660

Mathematics and Computers
TID-4500

Distribution:

Abbatiello, L. A.		Lang, D. M./Pasquier, L. P.	
Bailey, E. W.		Miskell, R. V.	
Ballenger, H. F.		Mitchel, G. W.	
Barbo, T. E.	(LASL)	McCoy, J. C.	(Bendix-KC)
Blackerby, Jack		Oliver, D. A.	(San-Alb)
Bookhart, T. W.	(20)	Radcliffe, J. N.	(Plastics Institute, London, England)
Dawson, Ray	(Black Clawson)		
Easterday, R. J.	(ORGDP)	Sestric, M. D.	(Bendix-KC)
Forte, N. A.	(GE-Evandale)	Smith, R. D.	
Foulk, D. L.		Spies, R. J.	(LRL)
Frye, E. R.	(San-Alb)	Stoner, H. H.	
Garner, B. L.	(LRL)	Technical Library	(San-Liv)(10)
Gilson, J. D.	(San-Liv)(10)	Thomas, R. A.	(GE-Waynesboro)
Gold, S. P.	(San-Liv)	Weaver, R. W.	(San-Liv)
Horde, G. W.	(2)	Wendolkowski, W. S.	(ORGDP)
Jasny, G. R.		Wood, D. C.	
Jerahov, G. E.	(Aerojet-Sac)	Y-12 Central Files	(10)
Knight, C. E.		Y-12 Central Files	(Route)
La Chance, R. L.	(San-Alb)	Y-12 Central Files	(Y-12RC)

In addition, this report is distributed in accordance with the category Mathematics and Computers, as given in the "USAEC Standard Distribution Lists for Unclassified Scientific and Technical Reports", TID-4500.

ABSTRACT

Fortran computer programs have been written that will determine the geodesic paths on an arbitrary surface of revolution. The programs can also determine the number of circuits of the geodesics necessary to produce a wrap of a specified thickness. This thickness can be for one geodesic or be the cumulative buildup of many geodesics. Once the geodesic paths are determined, thickness profile and helix angle plots are produced. In addition, routines are available for plotting the geodesic paths on the developed surface giving a two-dimensional picture of the paths on the surface.

CONTENTS

SUMMARY

Computer routines have been developed for computing a geodesic path on an arbitrary surface of revolution. This computation is accomplished by approximating the surface with a series of conical and cylindrical sections (approximating the contour of the surface by straight-line segments) and determining the geodesic path on each section. It was shown by S. P. Gold[1] that the geodesic on the approximated surface converges to the geodesic on the actual surface as the surface approximation converges.

In cylindrical coordinates, $r = kz + b$ on each of the sections, and a geodesic path can be written in terms of θ as a function of z. If the contour of the surface is approximated by straight lines joining the points (r_n, z_n) and the initial conditions of the geodesic are that it pass through the point (r_0, z_0, θ_0) at the helix angle α_0, the theta (mandrel) rotation (R_c) for one circuit is found to be:

$$R_c = 2 \sum_{n=J}^{L} \Delta\theta_n ,$$

where J and L are the sections in which the geodesic turns around and:

$$\Delta\theta_n = \theta\left(z_{n+1}\right) - \theta\left(z_n\right)$$

$$\Delta\theta_n = \begin{cases} \left(\sqrt{1+k_n^2}/k_n\right)\left[\sec^{-1}\left(r_{n+1}/c\right) - \sec^{-1}\left(r_n/c\right)\right] & \text{on conical sections } \left(k_n \neq 0\right),\ n \neq J, L, \\ \left(z_{n+1} - z_n\right)c \Big/ \left(r_n\sqrt{r_n^2 - c^2}\right) & \text{on cylindrical section } \left(k_n = 0\right), \\ \left(\sqrt{1+k_n^2}/k_n\right)\left[\sec^{-1}\left(r_{n+1}/c\right)\right] & \text{on section J,} \\ \left(\sqrt{1+k_n^2}/k_n\right)\left[0 - \sec^{-1}\left(r_n/c\right)\right] & \text{on section L,} \end{cases}$$

where:

$k_n = \left(r_{n+1} - r_n\right) / \left(z_{n+1} - z_n\right)$ section slope, and

$c = r_0 \sin\alpha_0$.

The geodesic turns around at the points where the surface radius equals c; that is,

$$r_{min} = c$$

$$= r_0 \sin \alpha_0 .$$

The value R_c (in radians) when divided by 2π is the number of revolutions per circuit for the specified geodesic. If R_c is written as a fraction, A/B, where A and B have no common factors, then A is the number of revolutions per pattern and B the number of circuits per pattern. A routine is included to find, if desired, a new value of α_0 which will produce a geodesic with a specified number of revolutions per circuit.

Computer routines have also been written to compute and plot two factors used in the stress analysis of filament-wound structures, helix angle, and thickness of wrap. In addition, the routines will determine the number of circuits necessary to produce a specified thickness at a point. This specified thickness can either be from one geodesic or be the cumulative buildup of many geodesics.

One of the useful by-products of approximating a surface by a series of conical and cylindrical sections is that cones and cylinders are developable; that is, if sliced, they can be laid out flat in a plane (see Appendix A). Geodesics on a cone or cylinder become straight lines on the developed surface. Routines, both Fortran and APT, have been written to draw the developed surface and to plot geodesic paths on this developed surface. This developed surface plot has been useful in determining certain characteristics such as thickness of wrap and number and location of crossovers. The developed surface plot can also be used to set up a winding machine by cutting out the plot and pasting it on the mandrel to be wrapped.

These routines are useful to engineers in designing wrap patterns for filament-wound structures. They are also the basis for routines used in locating the path of a filament feed eye of a numerically controlled filament-winding machine.(2)

INTRODUCTION

Combining high-strength filaments with resins in a composite structure has led to structural elements and parts which have exceedingly high strength-to-weight ratios. New materials, which lend themselves to filament windings, are being rapidly developed and new applications of composite structures are appearing. Products currently made by filament-winding techniques range from light-weight fishing rods to large railway tank cars.

As the applications of filament winding increase, so does the need for a better understanding and definition of wrapping patterns. One large class of filament-winding applications involves shapes which are surfaces of revolution. Since a geodesic path on any surface is a stable path, geodesics are often chosen as the desired filament paths. Therefore, this investigation was made by Y-12 Plant personnel to determine geodesic paths on an arbitrary surface of revolution and to compute fiber helix angle and thickness buildup which would result from wrapping these patterns. The project was sponsored by Sandia Livermore and carried out under Purchase Order ASB 92-1849.

DISCUSSION OF THE STUDY

FILAMENT PATH ON A SURFACE OF REVOLUTION

Since a geodesic on a surface is a stable path,(3) a filament laid along a geodesic will have no tendency to side slip. For this reason, geodesics are often chosen for the desired filament paths. However, the equations for geodesics on surfaces other than simple surfaces such as spheres, cones, and cylinders are not easily determined. Therefore, a method of approximating a geodesic on an arbitrary surface of revolution is undertaken.

To determine a geodesic on an arbitrary surface of revolution, first approximate the contour of the surface by a series of short, straight-line segments. When rotated about the axis of revolution, these line segments generate a series of conical and cylindrical sections that approximate the surface of revolution. Then, by using the equations for a geodesic on cones and cylinders and by determining the criteria for crossing from one section to another, a geodesic can be computed for the arbitrary surface of revolution.

GEODESIC ON A CONE

The problem associated with surfaces of revolution can be simplified by using cylindrical coordinates (r, z, θ). On a surface of revolution, r is a function of z and a point or curve on the surface can be defined in terms of two variables, z and θ.

To determine a geodesic on a cone, the property of the geodesic that is utilized is that between any two points on a surface, the path of minimum arc length is a geodesic. Therefore, to determine a geodesic between two points (Figure 1), it is necessary to find the curve which minimizes the following integral (arc length):

$$\int_{z_0}^{z_1} \sqrt{1 + (dr/dz)^2 + r^2\,(d\theta/dz)^2}\; dz. \tag{1}$$

A necessary condition(4,5) for the integral to be a minimum is:

$$\frac{d}{dz}\;\frac{r^2\,(d\theta/dz)}{\sqrt{1 + (dr/dz)^2 + r^2\,(d\theta/dz)^2}} = 0 \text{ , or:} \tag{2}$$

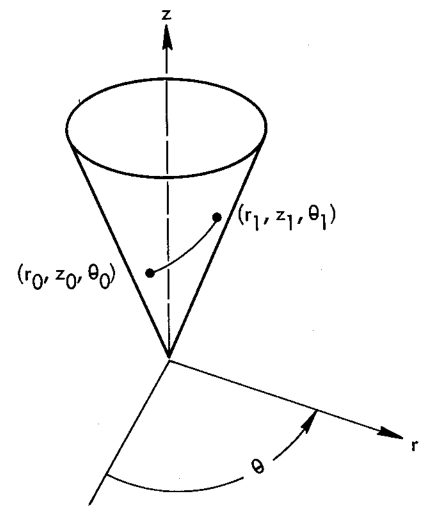

Figure 1. GEODESIC PATH ON A CONE.

$$\frac{r^2\,(d\theta/dz)}{\sqrt{1 + (dr/dz)^2 + r^2\,(d\theta/dz)^2}} = c = \text{constant of integration.} \tag{3}$$

Since the surface of revolution here is a cone, then:

$$r(z) = kz + b. \tag{4}$$

Equation 3 reduces to:

$$\frac{r^2(d\theta/dz)}{\sqrt{1 + k^2 + r^2\,(d\theta/dz)^2}} = c\ . \tag{5}$$

By squaring both sides and collecting terms, Equation 5 reduces to:

$$d\theta/dz = c\ \sqrt{1 + k^2}\ /\ \left(r\sqrt{r^2 - c^2}\right). \tag{6}$$

Solving Equation 6 results in:

$$\theta(z) = \left(\sqrt{1 + k^2}/k\right) \sec^{-1}\ \left[r(z)/c\right] + d. \tag{7}$$

If the geodesic passes through the point (r_0, z_0, θ_0) at helix angle α_0 (a common way of specifying the initial conditions for a geodesic), the constants c and d are found (Appendix B) to be:

$$c = r_0 \sin \alpha_0, \text{ and} \tag{8}$$

$$d = \left(\sqrt{1 + k^2}/k\right)\left[0 - \sec^{-1} (r_0/c)\right] + \theta_0 .$$

Thus, the equation for a geodesic on a cone is:

$$\theta (z) = \left(\sqrt{1 + k^2}/k\right)\left\{\sec^{-1} [r(z)/r_0 \sin \alpha_0] - (\pi/2 - \alpha_0)\right\} + \theta_0 . \tag{9}$$

In the special case of a cylinder, where $r \equiv r_0$, the differential equation is:

$$r_0^2 (d\theta/dz) \Big/ \sqrt{1 + r_0^2 (d\theta/dz)^2} = c \text{ , or:}$$

$$d\theta/dz = c \Big/ \left(r_0 \sqrt{r_0^2 - c^2}\right) .$$

The equation for the geodesic on a cylinder becomes:

$$\theta (z) = c \left(z - z_0\right) \Big/ \left(r_0 \sqrt{r_0^2 - c^2}\right) + \theta_0 ,$$

$$= (z - z_0) (1/r_0) \tan \alpha_0 + \theta_0 , \tag{10}$$

where, again:

$$c = r_0 \sin \alpha_0 .$$

SECTION-CROSSING CRITERIA

A surface composed of two cones is shown in Figure 2.

The equation for the surface is:

$$r(z) = k_1 (z - z_1) + r_1 \text{ for } z_0 \le z \le z_1 \text{, and}$$

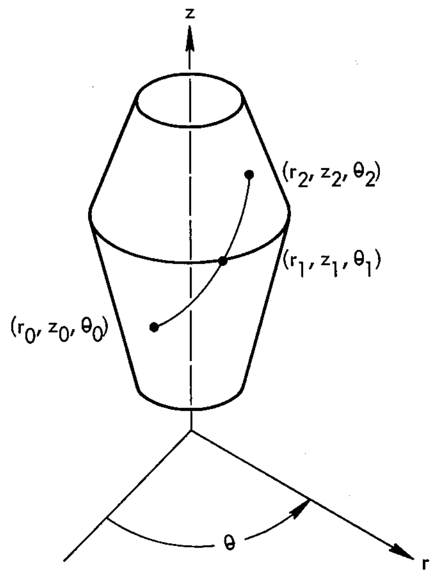

Figure 2. GEODESIC ON A SURFACE COMPOSED OF TWO CONES.

$$r\,(z) = k_2\,(z - z_1) + r_1 \text{ for } z_1 \leq z \leq z_2 .$$

From Equation 3 it is seen that the geodesic between (r_0, z_0, θ_0) and (r_1, z_1, θ_1) satisfies:

$$\frac{r^2\,(d\theta/dz)}{\sqrt{1 + r^2\,(d\theta/dz)^2 + (dr/dz)^2}} = c_1 ,$$

and satisfies:

$$\frac{r^2\,(d\theta/dz)}{\sqrt{1 + r^2\,(d\theta/dz)^2 + (dr/dz)^2}} = c_2 ,$$

between (r_1, z_1, θ_1) and $(r_2, z_2\, \theta_2)$.

The Weierstrass-Erdmann Corner Condition[4,5] is used to determine the necessary crossing condition for maintaining a geodesic on the composite surface; that is:

$$\lim_{z \to z_1^-} \frac{r^2\,(d\theta/dz)}{\sqrt{1 + r^2\,(d\theta/dz)^2 + (dr/dz)^2}} = \lim_{z \to z_1^+} \frac{r^2\,(d\theta/dz)}{\sqrt{1 + r^2\,(d\theta/dz)^2 + (dr/dz)^2}} .$$

Thus,

$$\begin{aligned} c_2 &= c_1 = c \\ &= r_0 \sin \alpha_0 . \end{aligned} \qquad (11)$$

(It is shown in Appendix B that the condition $c_2 = c_1$ implies that the helix angle is continuous at $z = z_1$.)

Then the equation for the geodesic is:

$$\theta(z) = \left(\sqrt{1 + k_1^2}/k_1\right) \left\{\sec^{-1} \left[r(z)/c\right] - (\pi/2 \quad \alpha_0)\right\} + \theta_0$$

for $z_0 \leq z \leq z_1$, and

$$\theta(z) = \left(\sqrt{1 + k_2^2}/k_2\right) \left\{\sec^{-1} \left[r(z)/c\right] - \sec^{-1}(r_1/c)\right\} + \theta_1$$

for $z_1 \leq z \leq z_2$, (12)

where:

$$\theta_1 = \theta(z_1) = \left(\sqrt{1 + k_1^2}/ k_1\right)\left[\sec^{-1} (r_1/ c) - (\pi/2 - \alpha_o)\right] + \theta_0,$$

and:

$$c = r_0 \sin \alpha_0 .$$

It was shown (Equation 6) that a geodesic on any cone, $r(z) = k_n z + b_n$, satisfies the differential equation:

$$d\theta/dz = c_n \sqrt{1 + k_n^2} / \left(r \sqrt{r^2 - c_n^2}\right) .$$

If the geodesic is a continuation of a geodesic which passed through the point (r_0, z_0, θ_0) at helix angle α_0, the constant of integration, c_n, is (see Equation 11):

$$c_n = c$$
$$= r_0 \sin \alpha_0 .$$

By rewriting the differential equation as:

$$dz/ d\theta = r \sqrt{r^2 - c^2} / \left(c \sqrt{1 + k_n^2}\right),$$

It is immediately seen that:

$$dz/d\theta\Big|_{r=c} = 0 \ .$$

Thus, the turnaround point of the geodesic is that location where the radius of the surface equals c (ie, equals $r_0 \sin \alpha_0$); that is, the radius at the turnaround is determined by r_0 and α_0 (radius and helix angle at the initial point) and is independent of the shape of the surface.

DETERMINING A CIRCUIT OF THE GEODESIC

Let the contour of the surface be defined by a series of straight-line segments joining the points (r_n, z_n), n = 1, . . ., M (Figure 3). For each segment, define the parameter k_n, as follows:

$$k_n = \left(r_{n+1} - r_n\right) / \left(z_{n+1} - z_n\right) \text{ for } n = 1, 2, \ldots, M - 1 \ .$$

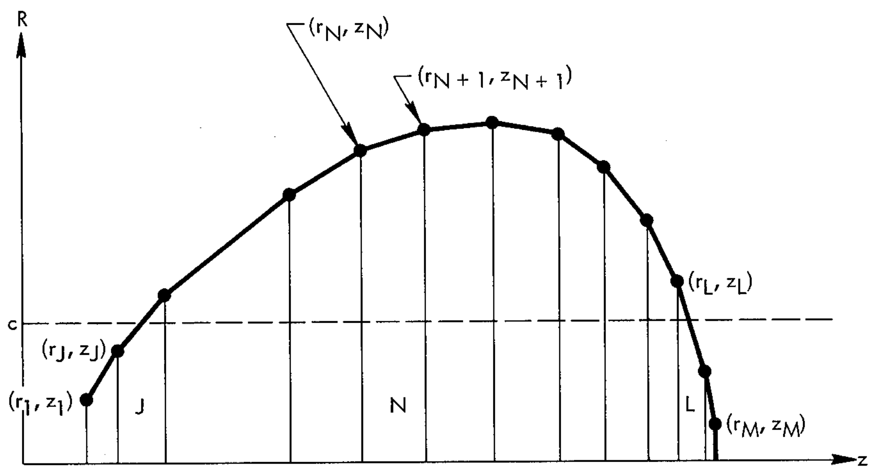

Figure 3. CONTOUR OF A SURFACE OF REVOLUTION.

If the initial conditions for specifying the geodesic are that the geodesic must pass through a point P_0, whose radius is r_0 at helix angle α_0, then the constant of integration, c, is:

$$c = r_0 \sin \alpha_0 \ .$$

Note here that c must be such that:

$$c \geq \left\{ \max r_1, r_M \right\} ;$$

otherwise, the geodesic would continue beyond the defined portion of the surface. Now, to determine the sections in which turnaround occurs, it is necessary to find J and L such that:

$$r_J \leq c < r_{J+1} \quad , \quad k_J > 0, \qquad \text{and}$$

$$r_L > c \geq r_{L+1} \quad , \quad k_L < 0.$$

Section J will be called the lower turnaround section and L the upper turnaround section. When $n \neq J,L$:

$$\Delta\theta_n = \theta\left(z_{n+1}\right) - \theta\left(z_n\right)$$

$$= \begin{cases} \left(\sqrt{1+k_n^2}\,/\,k_n\right)\left[\sec^{-1}\left(r_{n+1}/c\right) - \sec^{-1}\left(r_n/c\right)\right] \text{if } k_n \neq 0 \\ \qquad\qquad\qquad\qquad \text{(conical section)} \\ \left(z_{n+1} - z_n\right) c \,/\left(r_n \sqrt{r_n^2 - c^2}\right) \quad \text{if } k_n = 0. \\ \qquad\qquad\qquad\qquad \text{(cylindrical section)} \end{cases} \tag{13}$$

When n = J, L,

$$\Delta\theta_J = \left(\sqrt{1+k_J^2}\,/\,k_J\right)\left[\sec^{-1}\left(r_{J+1}/c\right) - 0\right], \quad \text{and} \tag{14}$$

$$\Delta\theta_L = \left(\sqrt{1+k_L^2}\,/k_L\right)\left[0 - \sec^{-1}\left(r_L/c\right)\right]. \tag{15}$$

The rotation during one circuit, R_c, becomes:

$$R_c = 2\left[\sum_{n=J}^{L} \Delta\theta_n\right]. \tag{16}$$

In order for the geodesic to return to its starting point (ie, complete one pattern), R_c (in revolutions) must be a rational number, say $R_c = A/B$. (In practice, R_c will always be rational since it is a computed value.) Then, after B circuits, the mandrel will have completed A revolutions and the geodesic will have returned to its starting point. If A and B have common factors, the geodesic will return to its starting point after fewer circuits. Thus, to determine when the path starts repeating, it is necessary to reduce A/B to a fraction which has no common factors. Once this is done, A becomes the number of revolutions per pattern and B the number of circuits per pattern.

Often the initial helix angle, α_0, is only an estimate of the desired helix angle at P_0. It may be more desirable to have a helix angle approximately equal α_0 at P_0, but which will produce a wrap having a predetermined number of circuits per pattern. This is the case when complete coverage is desired at a given parallel or where a certain thickness is wanted at a parallel. In Appendix B, an iterative scheme for choosing a new value for α_0 is derived to achieve the number of circuits per pattern.

GEODESIC ON A DEVELOPED SURFACE

One of the useful by-products of approximating a surface of revolution by a series of conical and cylindrical sections is that cones and cylinders are developable. That is, if sliced, they can be laid out flat in a plane. To further simplify matters, geodesics on a cone or cylinder become straight lines on the developed surface. Thus, a two-dimensional picture of a geodesic on the surface can be drawn.

Drawing a geodesic on a developed surface has been helpful in determining certain characteristics of a geodesic such as the thickness of the wrap and the number and location of the crossovers. The developed surface plot could also be used in setting up a winding machine by cutting out the plot and pasting it on the mandrel to be wrapped.

Computer routines have been written to compute a geodesic on a surface for given initial conditions, to develop the surface, and to plot the geodesic on the developed surface. As an example of this plot, geodesics were computed for the surface shown in Figure 4. The initial helix angles were adjusted (by the scheme discussed in Appendix B) so that the geodesic had 11 circuits per pattern (thus returning to its starting point after 11 circuits). Figure 5 shows a single geodesic on the surface, Figure 6 is the two-dimensional picture of the geodesic on the developed surface, and Figures 7 and 8 show the combined pattern of four geodesics on the surface.

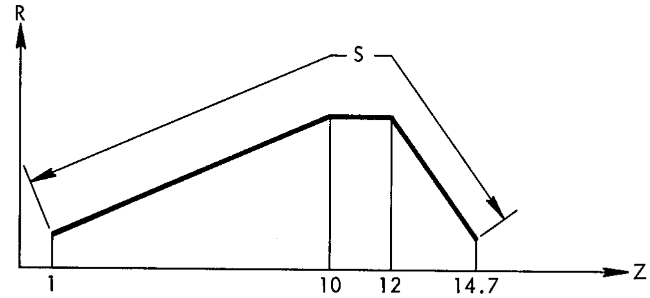

Figure 4. CONTOUR OF A SURFACE COMPOSED OF TWO CONICAL SECTIONS AND A CYLINDRICAL SECTION.

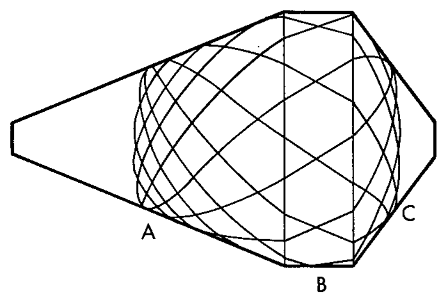

Figure 5. SURFACE WITH A SINGLE GEODESIC.

GEODESIC CHARACTERISTICS

Two parameters used in the stress analysis of a filament-wound structure are the helix angle and thickness of the wrap at various parallels. The helix angle can be determined directly from the relationships (Appendix B):

$$\tan \alpha = \left(c \,/\, \sqrt{r^2 - c^2} \right); \text{ that is,}$$

$$\alpha = \tan^{-1} \left(c \,/\, \sqrt{r^2 - c^2} \right), \tag{17}$$

where:

$$c = r_0 \sin \alpha_0 \, .$$

In determining the thickness of wrap at a given parallel, it is assumed that the center of the band follows the geodesic path. The approach used is to determine, at the desired parallel, the percentage of the circumference covered by a circuit of the geodesic. If the circuits are uniformly spaced around the part, then the computed percentage of coverage can be used to determine the average thickness at that parallel; that is:

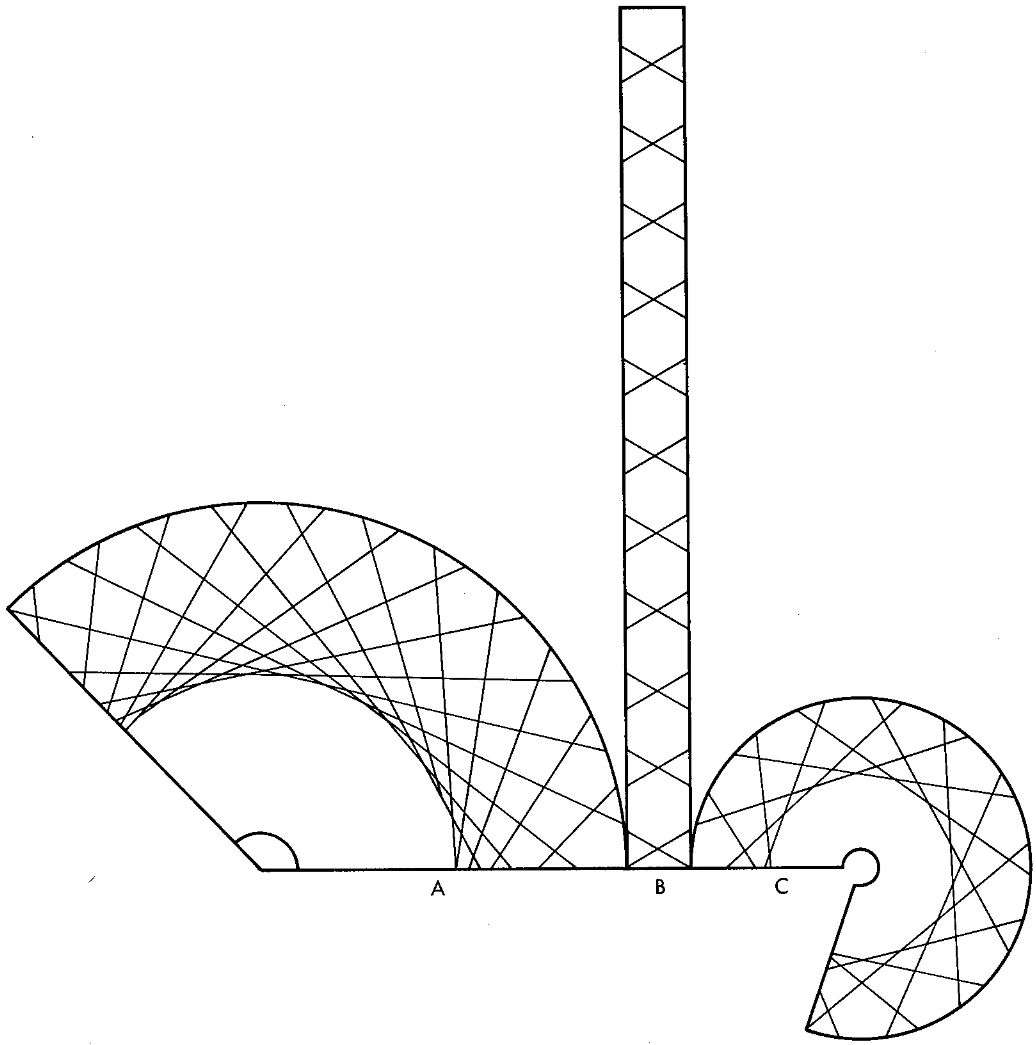

Figure 6. DEVELOPED SURFACE WITH A SINGLE GEODESIC.

$$\text{Average Thickness at a Parallel} = (\text{coverage/circuit})(\text{number of circuits})(\text{band thickness}). \tag{18}$$

For the derivation of the equations for coverage at a parallel, see Appendix C.

It should be noted here that the value computed for thickness is actually the amount of glass at the parallel. It does not take into account the matrix material present or the thickness resulting from voids and bridging of the fibers. Therefore, this figure should be modified by some factor determined by the percent glass of the wrap.

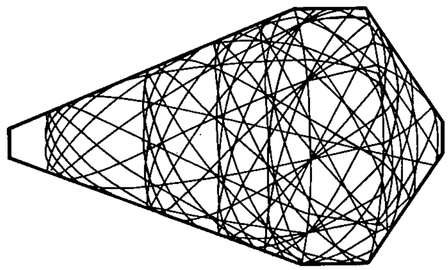

Figure 7. SURFACE WITH FOUR GEODESICS.

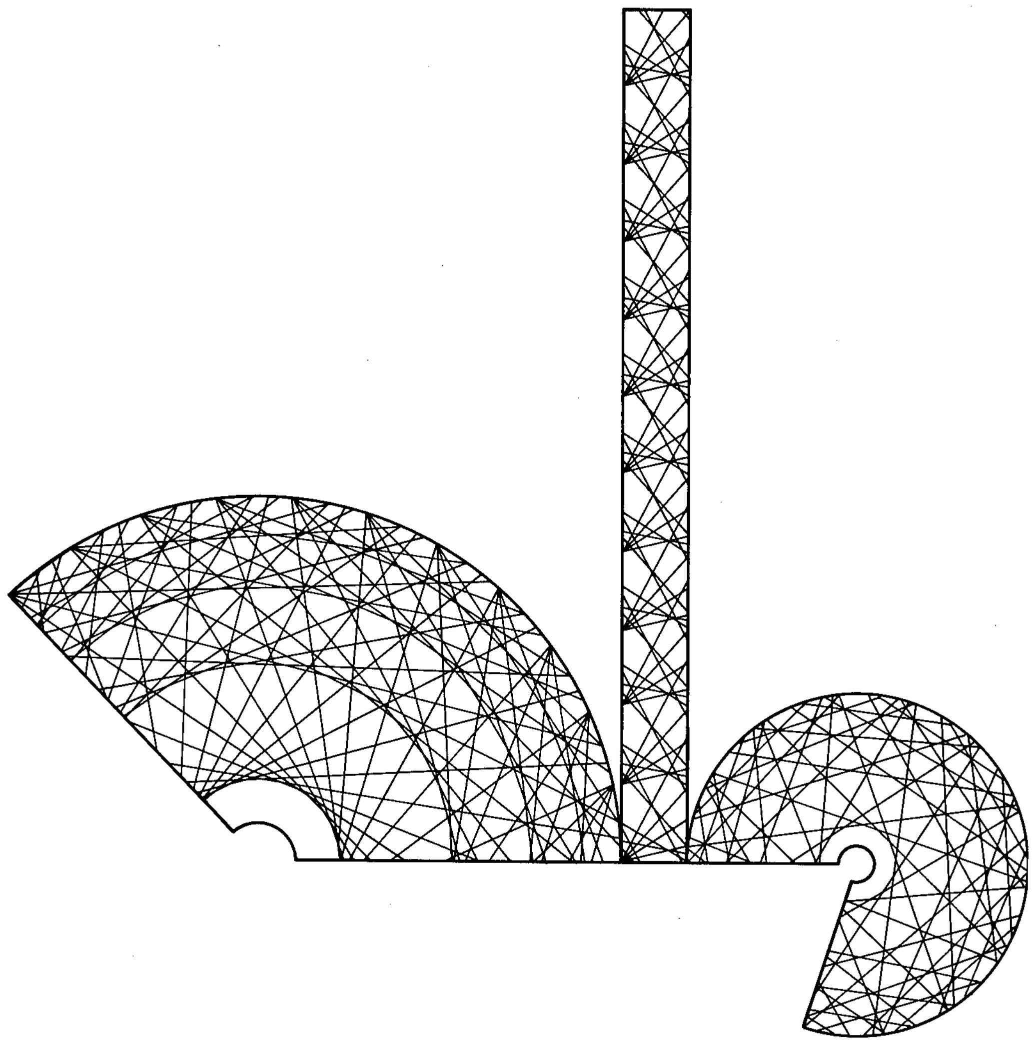

Figure 8. DEVELOPED SURFACE WITH FOUR GEODESICS.

For a given geodesic, Equation 18 can be used to determine the number of circuits necessary to build up a desired thickness at a parallel:

$$\text{Number of Circuits} = \frac{\text{(desired thickness at the parallel)}}{\text{(coverage/circuit at parallel)(band thickness)}} \; .$$

However, knowing the number of circuits to be wrapped does not fully describe the wrap pattern. It may be desirable to have these circuits uniformly spaced around the part. This possibility brings up an interesting question: Of how many patterns and circuits per pattern should the wrap consist? In trying to answer this question, two approaches are taken. They appear as options in the computer program (subroutine NOCIRC, described in Appendix D).

> Option 1 - When the surface to be wrapped is primarily a cylinder, it may be desirable to have the circuits spaced around the part so that after one pattern, the cylindrical portion is completely covered. Here, the number of circuits per pattern is chosen to give complete coverage at a parallel with no overlapping of fibers going in the same direction. The number of patterns necessary to build up the desired thickness is then determined.

> Option 2 - When wrapping a general surface of revolution, complete coverage at one parallel would produce overlapping fibers or less than complete coverage at all other parallels. Therefore, it is felt that the idea of complete coverage at a parallel has less meaning here. Also, in wrapping a general shape, it may be desirable to apply many different geodesics, building up a thin layer with each to achieve an overall wrap of a given thickness. The different geodesics could be chosen to produce this wrap. Thus, with this option, the number of circuits per pattern is chosen to equal the total number of circuits to be wrapped for the geodesic. Hence, after one pattern, the desired thickness for that geodesic is obtained.

To achieve a desired thickness at a parallel, the number of circuits per pattern, and number of patterns are determined by use of one of the two options. The desired thickness could be for this particular geodesic or the cumulative thickness of this and all prior geodesics. If it is the cumulative thickness that is wanted, then the thicknesses resulting from the previous geodesics are computed and subtracted from the thickness specified. This value is then used in determining the desired number of circuits. However, the number of circuits per pattern of the geodesic determined by the specified initial conditions will not, in general, be the same as those needed to give this wrap. Hence, it may be necessary to find a geodesic which differs slightly from the initially specified one, but which has the needed number of circuits per pattern.

The procedure for finding the new path is as follows: If A/B is the computed revolutions per circuit of the specified geodesic and NB the desired circuits per pattern, an integer NA is found so that NA/NB is as close as possible to A/B.

If NA and NB have common factors, NA and/or NB are altered so that there are no common factors. Then NA becomes the number of revolutions per pattern and NB the circuits per pattern. A new geodesic having NA/NB revolutions per circuit can then be found (by the scheme described in Appendix B) or the rotation of the computed geodesic can be distorted to achieve the desired revolutions per circuit.

Computer routines have been written for plotting these geodesic characteristics (helix angle and thickness). For plotting purposes, distance along the contour of the surface, S, was chosen as the reference (see Figure 4). For consistency and ease in plotting, all of the quantities are normalized before plotting.

Plots were made for the surface and geodesics shown in Figure 7. Values were computed for a 0.6-inch-wide band, 0.01 inch in thickness. The first plot, Figure 9, relates R and Z to the reference S; Figure 10 is a plot of the helix angles for the four geodesics. Figure 11 is the thickness plot for one geodesic (the geodesic shown in Figure 5), Figure 12 shows the thickness resulting from the four geodesics, and Figure 13 is a scale drawing of the contour after the wrap.

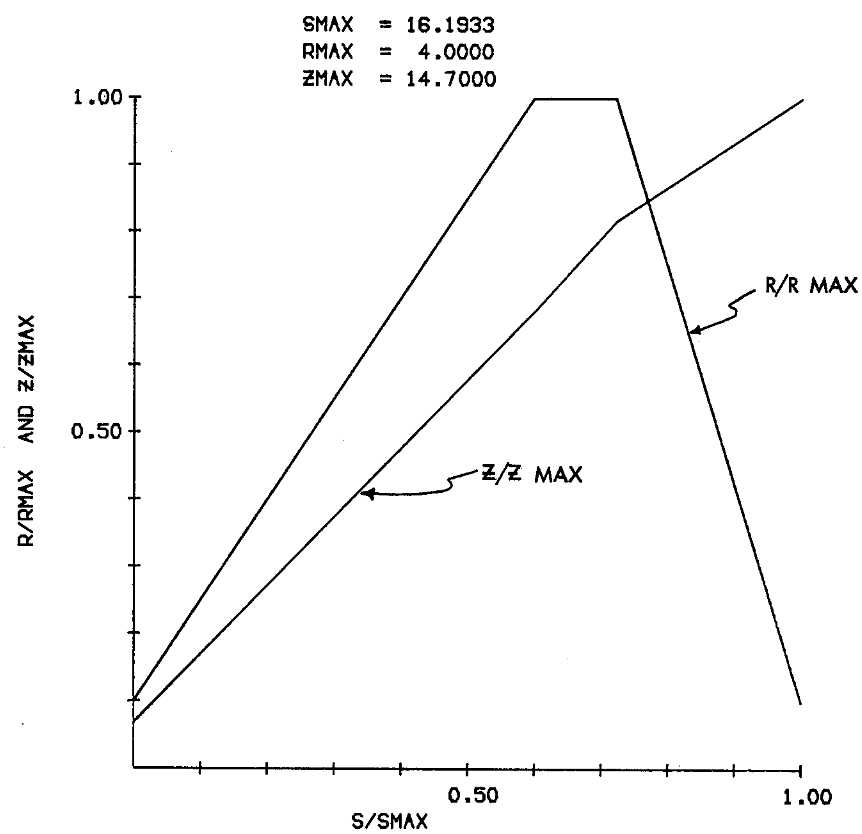

Figure 9. PLOT RELATING R AND Z (NORMALIZED) TO REFERENCE S.

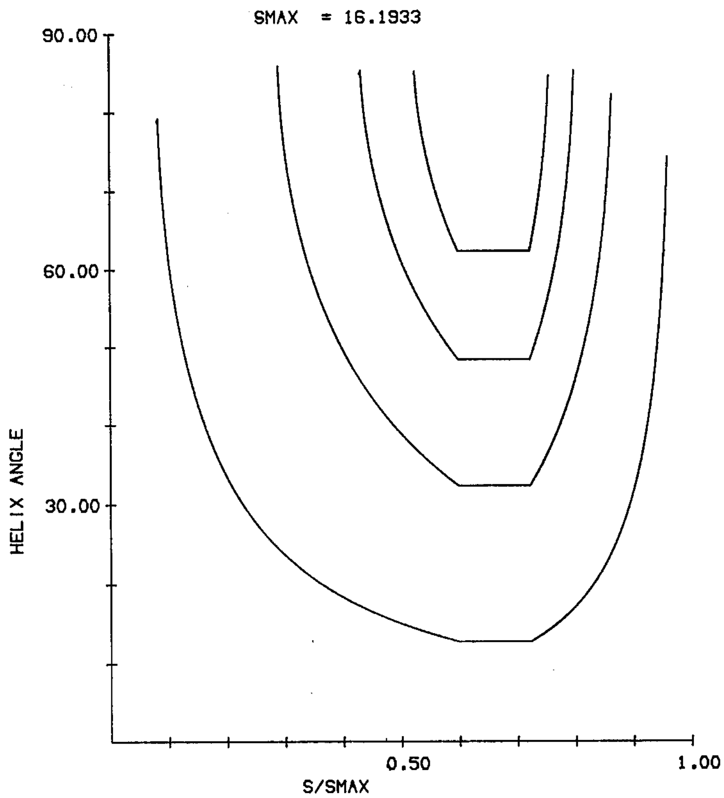

Figure 10. HELIX ANGLE PLOT FOR THE GEODESICS SHOWN IN FIGURE 7.

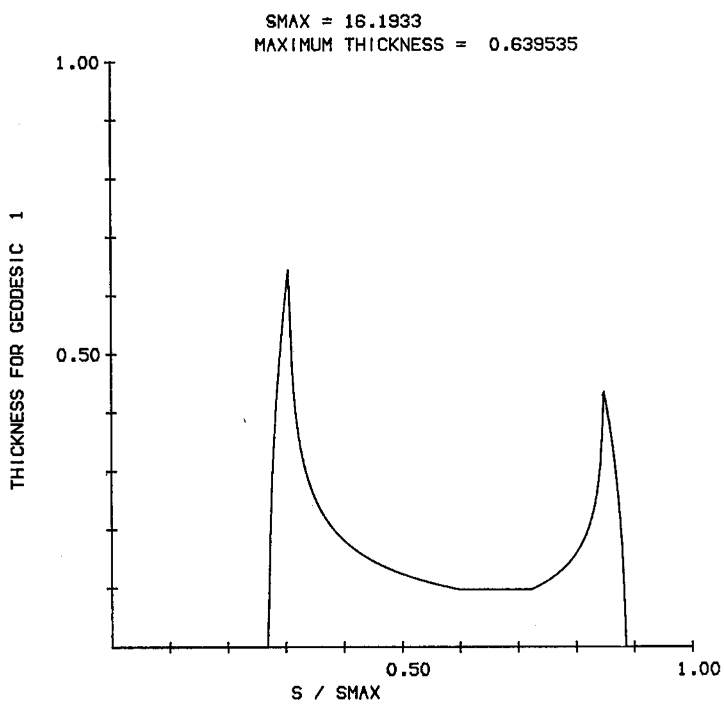

Figure 11. THICKNESS PLOT FOR THE GEODESIC SHOWN IN FIGURE 5.

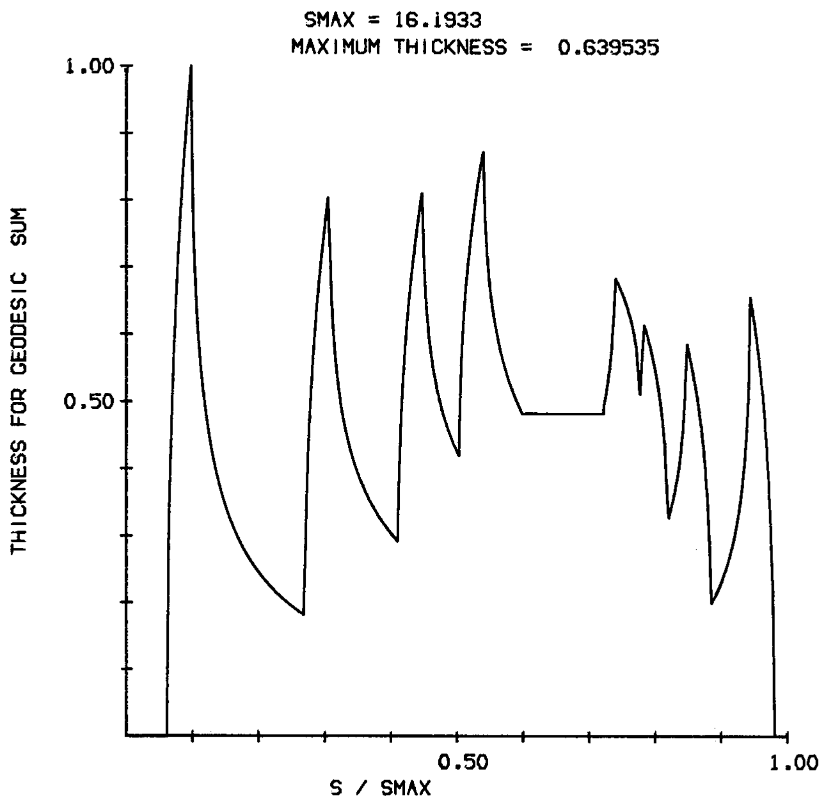

Figure 12. THICKNESS PLOT FOR THE GEODESICS SHOWN IN FIGURE 7.

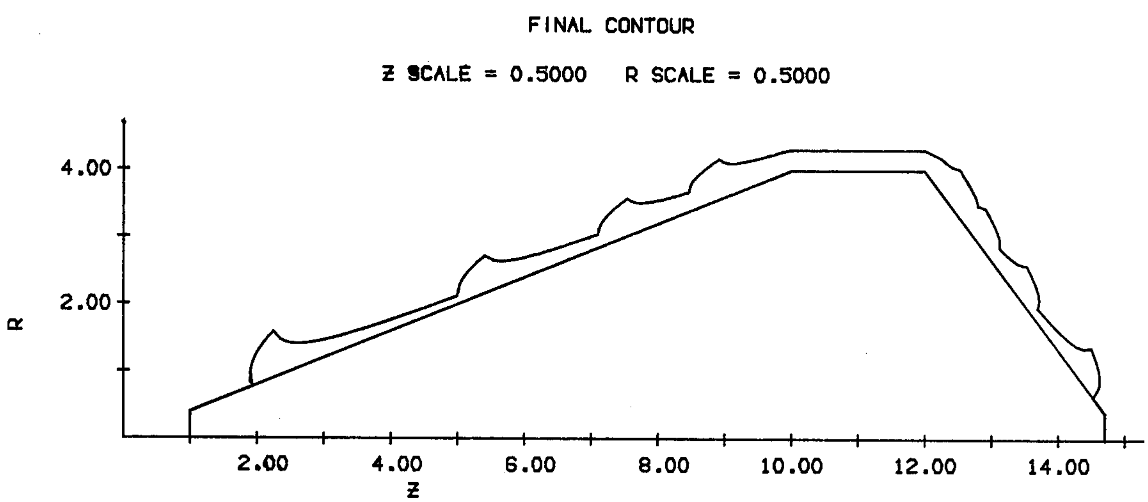

Figure 13. SCALED PLOT FOR THE GEODESICS SHOWN IN FIGURE 7.

COMPUTER PROGRAMS

The programs for computing a geodesic and plotting its characteristics are written in Fortran II. There are, in addition, four APT macros available for computing a geodesic and plotting it on the developed surface.

Fortran Program

The Fortran program consists of two main programs and 17 subroutines. In addition, the plotting routines utilize several subroutines for the Gerber Scientific Plotter.[6] With slight modification, the Gerber subroutines could be used with other plotting machines.

The geodesic subroutines are called by one of the main programs. Main program DESIGN is utilized when computing and plotting geodesic characteristics; main program DEVPLT is used for plotting geodesics on a developed surface. Flow sheets of the main program and deck arrangements for the two operations are shown in Appendix D. Also given in Appendix D are input details and a listing of the computer program.

APT Program

The APT program represents the initial efforts on this project. Due to the limited amount of storage available in APT, this approach was abandoned and the Fortran program undertaken. Therefore, the APT program, consisting of four macros, is limited to computing a geodesic and plotting it on the developed surface. These macros are briefly described in Appendix D.

COMPARISON OF A TRUE GEODESIC WITH A GEODESIC COMPUTED BY THE APPROXIMATION TECHNIQUE

The technique described in this report is the computation of a geodesic for a surface which is, in effect, an approximation of some other surface. A logical question to be raised is just how good does this computed path conform to a geodesic on the original surface? S. P. Gold proves that the path on the approximated surface converges to the geodesic on the true surface as the surface approximation converges.[1]

As an example of how well the approximation technique works, geodesics on a sphere were compared to those computed by the approximation technique. A filament will be on the mandrel surface even if a coarse approximation is used in calculating its path. For a given point (r, z, θ) on the filament path, there will be, for a given z, no error in r (since the point lies on the mandrel surface) between the

filament path and the true geodesic (great circle). The deviation, if any, will be in the rotation, θ. Therefore, in comparing the computed path with the great circle, the rotation for a great circle (360°) is compared with the rotation as computed.

Geodesics were computed for six approximations of the sphere. These approximations ranged from 18 conical sections (19 equally spaced points on the sphere) to an approximation involving 720 sections. Geodesics with helix angles (at the equator of the sphere) of 10 to 85 degrees were determined. The results are summarized in Table 1. It can be seen from this table that the approximation technique determines a path which closely follows the true geodesic on a sphere. The finer the approximation of that portion of the sphere on which the geodesic travels, the smaller the deviation between the great circle and the computed path.

Table 1

COMPARISON OF THE ROTATION OF A TRUE GEODESIC ON A SPHERE WITH GEODESICS ON VARIOUS APPROXIMATIONS OF THE SPHERE

Helix Angle (degrees)	Number of Conical Sections Approximating a Sphere	Number of Sections Traversed by the Geodesic	Rotation for Circuit (degrees)	Deviation per Circuit (degrees)	Percent Deviation
40	18	10	366.653	+6.653	1.85
65	36	10	366.433	+6.433	1.79
80	90	10	366.342	+6.342	1.76
85	180	10	366.331	+6.331	1.76
40	36	20	362.244	+2.244	0.622
70	90	20	362.196	+2.196	0.612
80	180	20	362.184	+2.184	0.607
85	360	20	362.185	+2.185	0.607
50	90	40	360.773	+0.773	0.215
70	180	40	360.765	+0.765	0.212
80	360	40	360.764	+0.764	0.212
85	720	40	360.770	+0.770	0.214
10	180	80	360.269	+0.269	0.075
70	360	80	360.267	+0.267	0.074
80	720	80	360.268	+0.268	0.074
10	180	160	360.071	+0.071	0.020
50	360	160	360.094	+0.094	0.026
70	720	160	360.093	+0.093	0.026

DEFINITION OF TERMS

Geodesics - A path is called a geodesic on a surface if at each point of the path, the principal normal coincides with the normal to the surface. (The shortest of all paths joining two points on a surface is an arc of a geodesic.)[3]

Meridian - Any plane which passes through the axis of revolution intersects a surface of a revolution along a pair of curves. The curves are called meridians.

Helix Angle - If P is a point of a geodesic on a surface of revolution, then the angle between the geodesic and the meridian at point P is the helix angle at P.

Parallel - Every plane perpendicular to the axis of revolution intersects a surface of revolution along a circle, which is called a parallel.

Circuit - The path traced from a starting point at a particular parallel on a surface until the path crosses the same parallel going in the same direction is one circuit.

Pattern - The number of circuits the path traces on a surface in returning to its original starting point is a pattern.

APPENDIX A

DEVELOPED SURFACE

Developing a Surface

Let the contour of a surface be defined as a series of straight-line segments joining the points (r_n, z_n), n = 1, M. Define the following section parameters:

$$k_n = \left(r_{n+1} - r_n\right) / \left(z_{n+1} - z_n\right) \qquad n = 1, 2, \ldots, M-1$$

$$f_n = \sqrt{1 + k_n^2} \qquad n = 1, 2, \ldots, M-1$$

$$x_1 = 0$$

$$x_{n+1} = x_n + \left(z_{n+1} - z_n\right) f_n \qquad n = 1, 2, \ldots, M-1$$

$$\varphi_n = \left| k_n / f_n \right| \; 2\pi \qquad n = 1, 2, \ldots, M-1$$

$$R1_n = \left| f_n / k_n \right| r \qquad r = \begin{cases} r_n \text{ if } k_n > 0 \\ r_{n+1} \text{ if } k_n < 0 \end{cases} \qquad n = 1, 2, \ldots, M-1$$

$$R2_n = \left| f_n / k_n \right| r \qquad r = \begin{cases} r_{n+1} \text{ if } k_n > 0 \\ r_n \text{ if } k_n < 0 \end{cases} \qquad n = 1, 2, \ldots, M-1$$

$$xc_n = \begin{cases} x_n - R1_n & \text{if } k_n > 0 \\ x_n + R2_n & \text{if } k_n < 0 \end{cases} \qquad n = 1, 2, \ldots, M-1$$

Utilizing these parameters, the surface can be developed. Figure A-1 is an example of a surface which has been developed.

Tranformation of a Point on Surface (z, θ) to a Point on the Developed Surface (x, y)

1. Find n such that:

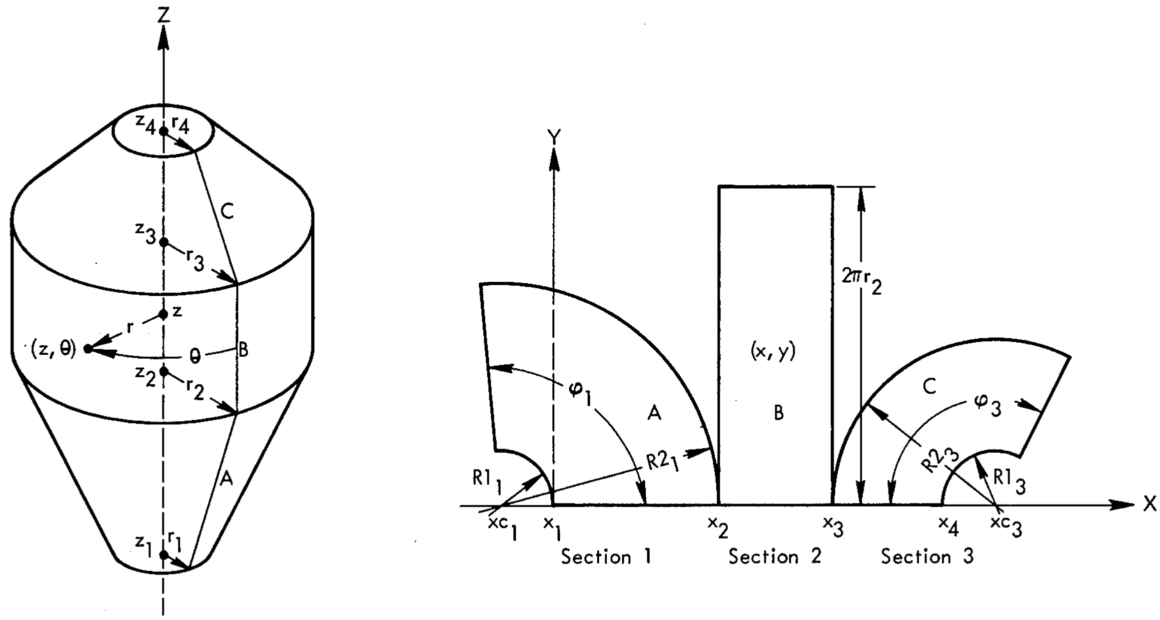

Figure A-1. A DEVELOPED SURFACE COMPOSED OF TWO CONICAL SECTIONS AND A CYLINDRICAL SECTION.

$$z_n \leq z \leq z_{n+1} \quad .$$

2. If $k_n = 0$ (cylindrical section), then:

$$x = x_n + \left(z - z_n\right), \text{and} \tag{19}$$

$$y = r_n \, \theta \quad . \tag{20}$$

3. If $k_n \neq 0$ (conical section), then:

$$x = \begin{cases} xc_n + \left(f_n / k_n\right) \; r \cos\left[\left(k_n / f_n\right) \theta\right] & \text{if } k_n > 0 \\ xc_n - \left|f_n / k_n\right| \; r \cos\left[\left(k_n / f_n\right) \; \theta\right] & \text{if } k_n < 0, \end{cases}$$

$$y = \left| f_n / k_n \right| \; r \; \sin\left(\left| k_n / f_n \right| \theta\right) \quad .$$

These equations reduce to:

$$x = x_n - \left(f_n / k_n\right)\left\{r_n - \left[r_n + k_n\left(z - z_n\right)\right] \cos\left[\left(k_n / f_n\right)\theta\right]\right\}, \tag{21}$$

$$y = \left(f_n / k_n\right)\left[r_n + k_n\left(z - z_n\right)\right] \sin\left[\left(k_n / f_n\right)\theta\right]. \tag{22}$$

Drawing a Geodesic on a Developed Surface

Drawing a geodesic on a conical section ($k_n > 0$) will be discussed. Since the other cases ($k_n \leq 0$) are similar, they will not be presented. Let the geodesic enter section n at (r, z, θ) either initially or by transition from another section. Define:

$$\beta_0 = \left(k_n / f_n\right)\theta,$$

$$\Delta\beta = \left(k_n / f_n\right)\Delta\theta_n,$$

$$\rho_1 = \begin{cases} R2_n & \text{if previous section was section } n + 1 \\ R1_n & \text{if previous section was section } n - 1 \end{cases},$$

$$\rho_2 = \begin{cases} R1_n & \text{if } \rho_1 = R2_n \\ R2_n & \text{if } \rho_1 = R1_n \end{cases}.$$

Case A-1 — $\beta_0 + \Delta\beta < \varphi_n$, when ($n \neq J$) (Figure A-2) - The geodesic is the line segment between (x_e, y_e) and (x_d, y_d), where:

$$x_e = xc_n + \rho_1 \cos\beta_0,$$

$$y_e = \rho_1 \sin\beta_0,$$

$$x_d = xc_n + \rho_2 \cos(\beta_0 + \Delta\beta),$$

$$y_d = \rho_2 \sin(\beta_0 + \Delta\beta).$$

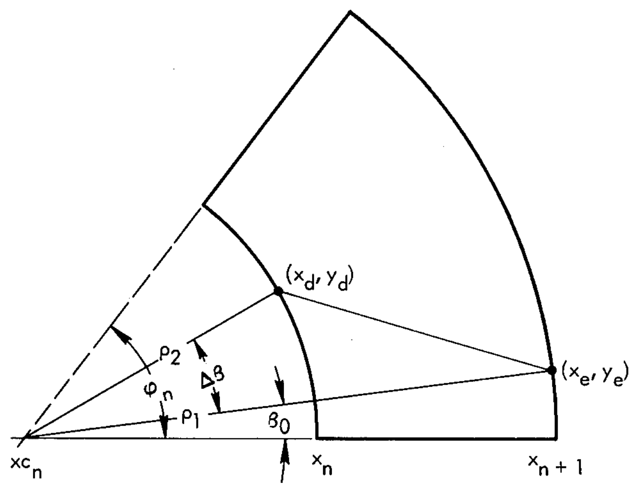

Figure A-2. GEODESIC ENTERING AND LEAVING A CONICAL SECTION. (Line on the Developed Surface)

Case A-2 — $\beta_0 + \Delta\beta > \varphi_n$, when ($n \neq J$) (Figure A-3) - In this case, the geodesic is represented as two line segments, from (x_e, y_e) to (x_1, y_1) and from (x_2, y_2) to (x_d', y_d'), as shown in Figure A-3. With respect to a local origin at $(xc_n, 0)$, the line through (x_e, y_e) has the equation:

$$y - y_e = \left(y_d - y_e\right)\left(x - x_e\right) / \left(x_d - x_e\right),$$

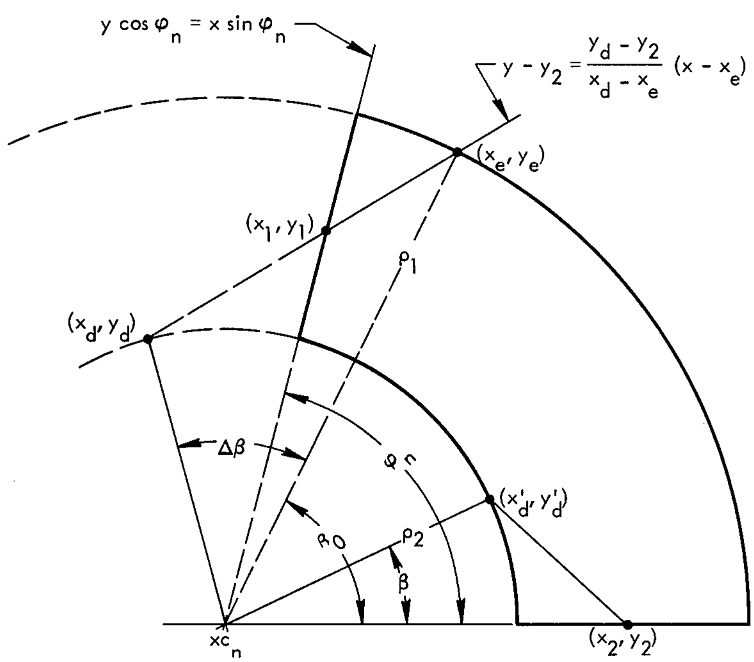

Figure A-3. GEODESIC ON TWO LINE SEGMENTS ON A DEVELOPED CONICAL SECTION.

and the line defining the end of the developed surface satisfies equation:

$$y \cos \varphi_n = x \sin \varphi_n ,$$

where:

$$x_e = \rho_1 \cos \beta_0,$$

$$y_e = \rho_1 \sin \beta_0,$$

$$x_d = \rho_2 \cos \left(\beta_0 + \Delta\beta\right),$$

$$y_d = \rho_2 \sin \left(\beta_0 + \Delta\beta\right).$$

Solving for the intersection of the two lines determines the point (x_1, y_1). Then (x_2, y_2) is found by:

$$x_2 = x_1^2 + y_1^2 ,$$

$$y_2 = 0 .$$

The end point of the line segment is:

$$x_d' = \rho_2 \cos \beta ,$$

$$y_d' = \rho_2 \sin \beta ,$$

where:

$$\beta = \beta_0 + \Delta\beta - \varphi_n .$$

Translating the points by xc_n locates the geodesic on the developed surface.

<u>Case A-3 — $\beta_0 + 2\Delta\beta \leq \varphi_n$, when $(n = J)$ (Figure A-4)</u> – The geodesic on Section J (turnaround section) is again a line segment between the points (x_e, y_e) and (x_d, y_d), as presented in Figure A-4, where:

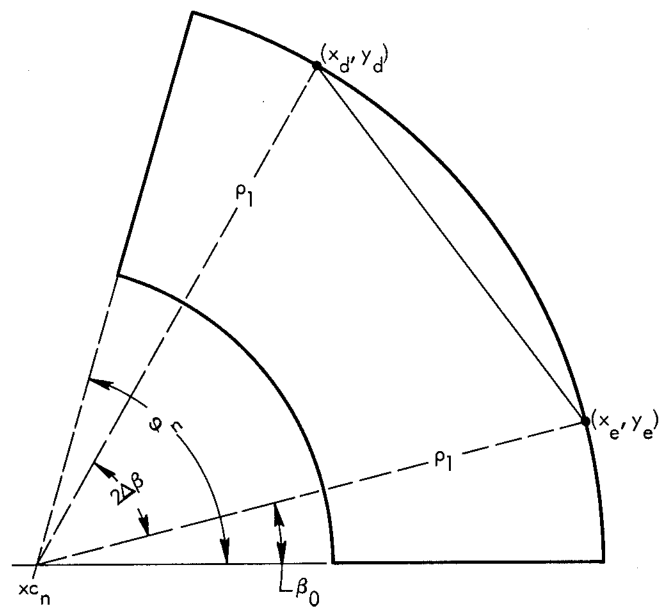

Figure A-4. GEODESIC ON A TURNAROUND SECTION OF A DEVELOPED SURFACE.

$$x_e = xc_n + \rho_1 \cos \beta_0 ,$$

$$y_e = \rho_1 \sin \beta_0 ,$$

$$x_d = xc_n + \rho_1 \cos \left(\beta_0 + 2\Delta\beta\right) ,$$

$$y_d = \rho_1 \sin \left(\beta_0 + 2\Delta\beta\right) .$$

The case, $\beta_0 + 2\,\Delta\beta > \varphi_n$, is similar to Case A-2 and can be determined in a similar fashion.

APPENDIX B

ADDITIONAL GEODESIC DERIVATIONS

Evaluation of the Constants of Integration

Let the initial conditions for specifying a geodesic be that the geodesic passes through point (r_0, z_0, θ_0) at helix angle α_0. To determine the constant of integration (c in Equation 3 of the text), two cases will be examined.

Case B-1 — Initial Point Lies in a Cylindrical Section - It was shown in Appendix A that in transforming a surface point on a cylinder (r, z, θ) to a point on the developed surface (x, y), the relationship is:

$$x = z + h, \text{ where h is a constant, and}$$

$$y = r\theta .$$

Then:

$$dy / dx = (dy / d\theta) / (dx / d\theta) ,$$

$$= r (d\theta / dz) , \text{ or}$$

$$d\theta / dz = (1/r) (dy/dx) .$$

But,

$$dy / dx = \tan \alpha_0, \text{ and}$$

$$d\theta/dz = (1/r) \tan \alpha_0, \text{ as shown in Figure B-1.}$$

The equation developed for the geodesic on the cylinder was determined to be:

$$r^2 (d\theta / dz) / \sqrt{1 + r^2 (d\theta / dz)^2} = c .$$

Therefore,

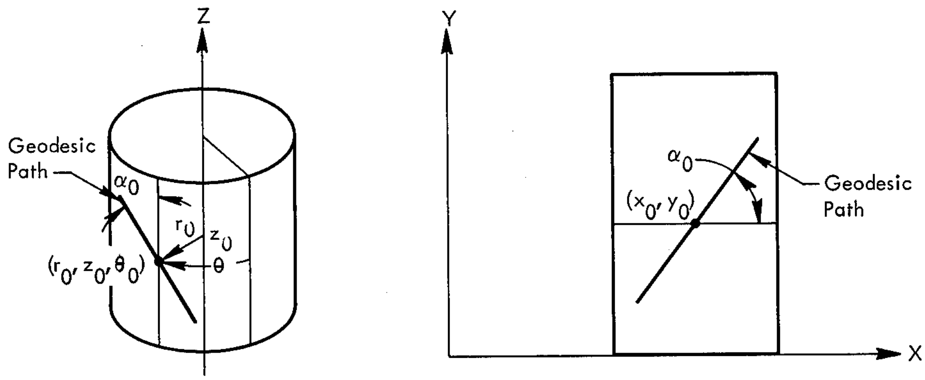

Figure B-1. DEVELOPED CYLINDRICAL SECTION.

$$c = r^2 (\tan \alpha_0 / r) / \sqrt{1 + r^2 \left(\tan^2 \alpha_0 / r^2\right)}, \text{ and}$$

$$c = r \sin \alpha_0 .$$

Since the initial point lies on the cylinder,

$$r \equiv r_0 , \text{ and}$$

$$c = r_0 \sin \alpha_0 .$$

Case B-2 — Initial Point in a Conical Section - As was true for the cylindrical case, the results of Appendix A (developing the surface) will be utilized here. The following relations are derived from Equation 21 and 22 of Appendix A:

$$dy/dx = \frac{\left[r_n + k_n \left(z - z_n\right)\right] \cos \left(k_n \theta / f_n\right) + (dz/d\theta)\, f_n \sin \left(k_n \theta / f_n\right)}{-\left[r_n + k_n \left(z - z_n\right)\right] \sin \left(k_n \theta / f_n\right) + (dz/d\theta)\, f_n \cos \left(k_n \theta / f_n\right)} ; \quad (23)$$

$$d\theta/dz = \left[\frac{f_n}{r_n + k_n \left(z - z_n\right)}\right] \frac{(dy/dx) \cos \left(k_n \theta / f_n\right) - \sin \left(k_n \theta / f_n\right)}{\cos \left(k_n \theta / f_n\right) + (dy/dx) \sin \left(k_n \theta / f_n\right)} . \quad (24)$$

By examining the initial conditons for $\theta_0 = 0$ (no generality lost here since a substitution $\theta = \theta - \theta_0$ would result in the same geodesic shifted by θ_0), the initial point would appear as shown in Figure B-2. Again,

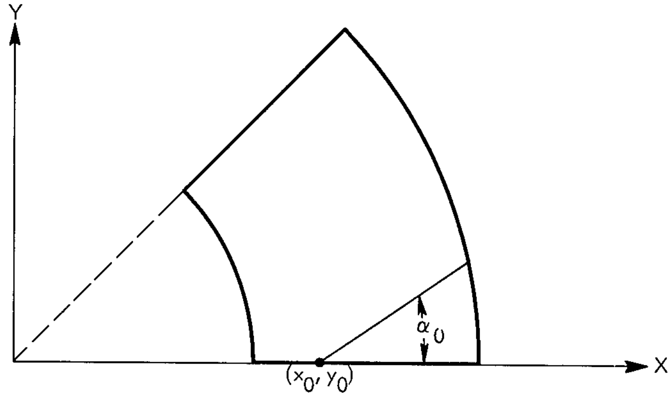

Figure B-2. DEVELOPED CONICAL SECTION.

$$dy/dx = \tan \alpha_0 \text{ , and}$$

$$\left. d\theta/dz \right|_{\left(r_0,\ z_0,\ 0\right)} = f_n \tan \alpha_0 \Big/ \left[r_n + k_n \left(z_0 - z_n \right) \right] ,$$

$$= f_n \tan \alpha_0 / r_0 \ .$$

The geodesic on the conical section satisfies (see Equation 5):

$$c = r^2 \, (d\theta/dz) \Big/ \sqrt{1 + k_n^2 + r^2 \, (d\theta/dz)^2} \ .$$

The constant, evaluated at $(r_0,\ z_0,\ 0)$, becomes:

$$c = r_0^2 \left(f_n \tan \alpha_0 / r_0 \right) \Big/ \sqrt{1 + k_n^2 + r_0^2 \left(f_n \tan \alpha_0 / r_0 \right)^2} \ ;$$

$$c = r_0 \sin \alpha_0 \ .$$

The constant d is found to be:

$$\theta \, (z_0) = \theta_0 = \left(\sqrt{1 + k^2} / k \right) \sec^{-1} \left(r_0 / c \right) + d, \text{ or}$$

$$d = \theta_0 - \left(\sqrt{1 + k^2}/k\right) \sec^{-1}\left(r_0/c\right).$$

Helix Angle at a Parallel

To determine the helix angle at a given parallel, the developed surface will again be utilized. By examining the geodesic at $\theta = 0$ (again no generality is lost), it is seen that:

$$dy/dx\Big|_{\theta = 0} = \tan\alpha .$$

For a conical section (see Equations 6 and 23),

$$dy/dx\Big|_{\theta = 0} = \left[r_n + k_n\left(z - z_n\right)\right] / \left[f_n \quad (dz/d\theta)\right],$$

$$= \left(r/f_n\right) d\theta/dz ,$$

$$= \left(r/f_n\right)\left(cf_n/r \sqrt{r^2 - c^2}\right),$$

$$= c/\sqrt{r^2 - c^2}.$$

Thus,

$$\tan\alpha = c/\sqrt{r^2 - c^2}, \text{ or} \tag{25}$$

$$\alpha = \tan^{-1}\left(c/\sqrt{r^2 - c^2}\right). \tag{26}$$

Using a similar argument, the same result can be derived for a cylindrical section.

Equation 25 can be rewritten as:

$$\sin\alpha = c/r , \text{ or}$$

$$r \sin\alpha = c. \tag{27}$$

Equation 27 is the same relationship given by Clairauts' Theorem[3] for a geodesic on a surface of revolution.

Since r is a continuous function and the constant c has the same value on each section, Equation 27 implies that the helix angle, α, is continuous at the point of transition from one section to another.

Determining an Initial Helix Angle to Produce a Geodesic with the Desired Number of Revolutions per Circuit

Previously, an equation for determining the rotation for a circuit of the geodesic, R_c, was derived. This was given by Equation 16 which is repeated below:

$$R_c = 2\left[\sum_{n=J}^{L} \Delta\theta_n\right],$$

where:

$$\Delta\theta_J = \left(f_J / k_J\right)\left[\sec^{-1}\left(r_{J+1}/c\right)\right],$$

$$\Delta\theta_L = \left(f_L / k_L\right)\left[-\sec^{-1}\left(r_L/c\right)\right],$$

$$\Delta\theta_n = \begin{cases} \left(f_n / k_n\right)\left[\sec^{-1}\left(r_{n+1}/c\right) - \sec^{-1}\left(r_n/c\right)\right] & \text{if } k_n \neq 0 \\ \left(z_{n+1} - z_n\right) c / \left(r_n \sqrt{r_n^2 - c^2}\right) & \text{if } k_n = 0 \end{cases},$$

$$c = r_0 \sin\alpha_0 .$$

If the desired rotation per circuit is $\overline{R}_c$ (to give complete coverage or a desired thickness, etc), define:

$$\Delta R_c = \overline{R}_c - R_c .$$

An approximation of an initial helix angle, $\overline{\alpha}_0$, which will produce a geodesic having the desired rotation per circuit is found by:

$$\Delta R_c / \Delta\alpha_0 \approx dR_c / d\alpha_0 ,$$

$$\Delta\alpha_0 \approx \Delta R_c / \left(dR_c / d\alpha_0 \right) ,$$

$$\overline{\alpha} = \alpha_0 + \Delta\alpha_0 .$$

Now,

$$dR_c / d\alpha_0 = d \left\{ 2 \left[\sum_{n=J}^{L} \Delta\theta_n \right] \right\} / d\alpha_0 ,$$

$$= 2 \sum_{n=J}^{L} \left[d\,\Delta\theta_n / d\,\alpha_0 \right] .$$

The derivatives are found to be:

$$d\,\Delta\theta_J / d\,\alpha_0 = -\left(f_J / k_J\right) r_0 \cos\alpha_0 / \sqrt{r_{J+1}^2 - c^2} ,$$

$$d\,\Delta\theta_L / d\alpha_0 = \left(f_L / k_L\right) r_0 \cos\alpha_0 / \sqrt{r_L^2 - c^2} ,$$

$$d\Delta\theta_n / d\alpha_0 = \begin{cases} \left(f_n r_0 \cos\alpha_0 / k_n\right) \left[-1/\sqrt{r_{n+1}^2 - c^2} + 1/\sqrt{r_n^2 - c^2} \right] & \text{if } k_n \neq 0 \\ r_0 \cos\alpha_0 \left[r_n \left(z_{n+1} - z_n\right) / \left(r_n^2 - c^2\right)^{3/2} \right] & \text{if } k_n = 0 \end{cases} .$$

APPENDIX C

DERIVATION OF EQUATIONS FOR THE THICKNESS OF WRAP

Computing the Coverage

As stated previously, the approach used in determining the thickness of wrap at a parallel is to find the fraction of the circumference which is covered by one circuit of the geodesic. In computing this coverage, it is assumed that the center of the band lies along the geodesic. The developed surface is utilized in each of the four cases considered below.

Case C-1 — Parallel in a Cylindrical Section - Each circuit of a geodesic will cross the parallel in a cylindrical section twice as shown in Figure C-1. The fraction of the circumference covered by each circuit is:

$$\text{COVERAGE/CIRCUIT} = 2(w/\cos\alpha)\,/\,2\pi r$$

$$= w/\pi r \cos\alpha,$$

where:

w represents the band width,

r the radius of cylindrical section, and

α the helix angle.

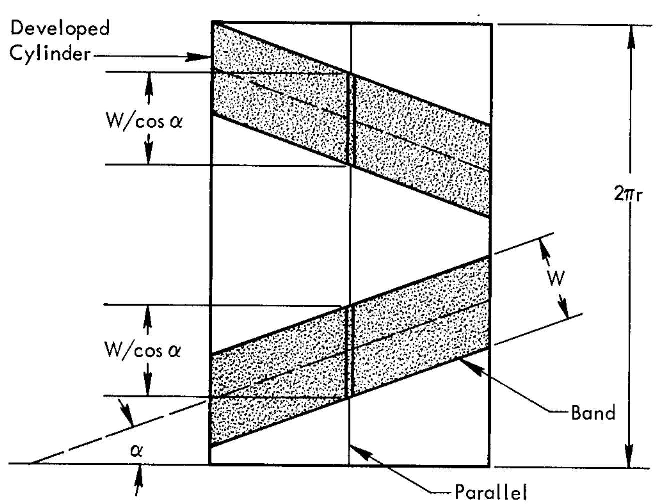

Figure C-1. DEVELOPED CYLINDER.

Case C-2 — Parallel in a Conical Section, Band Crosses Parallel Twice - In a conical section, a parallel is represented as a portion of a circle on the developed surface as shown in Figure C-2. To determine the fraction covered at the parallel, a reference frame is established with the origin at the intersection of the band center line and the parallel circle (see Figure C-3). The fraction of the developed cone angle covered is then determined.

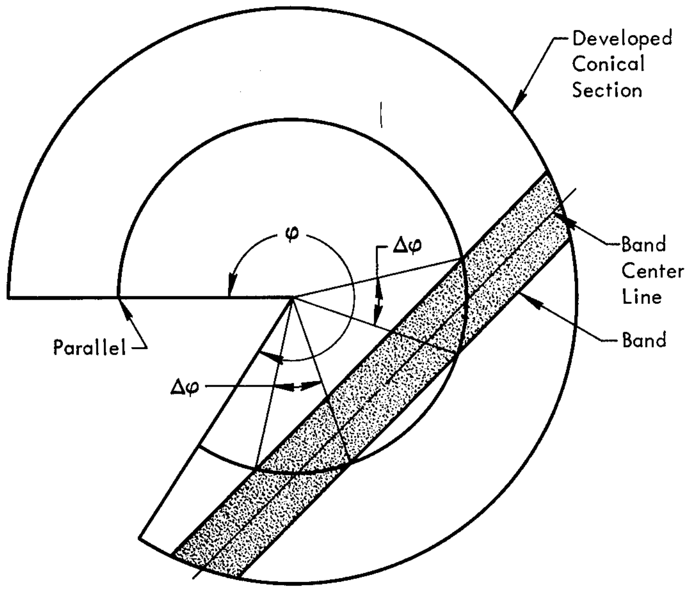

Figure C-2. DEVELOPED CONICAL SECTION. (Band Crossing Parallel Twice)

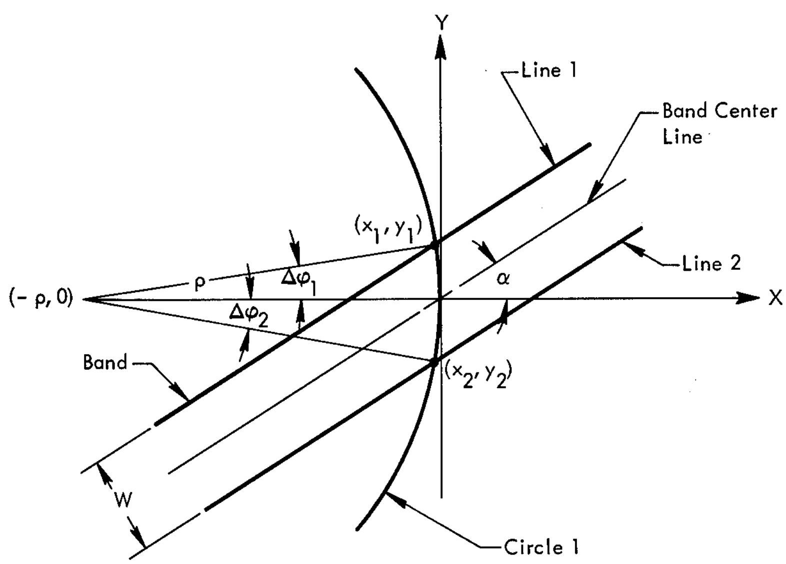

Figure C-3. COORDINATE FRAME WITH THE ORIGIN AT THE INTERSECTION OF THE BAND CENTER LINE AND PARALLEL CIRCLE.

The equations for the band and the parallel circle are:

$$\text{Line 1 -}\quad (-\sin\alpha)\,x + (\cos\alpha)\,y = w/2;$$

$$\text{Line 2 -}\quad (-\sin\alpha)\,x + (\cos\alpha)\,y = -w/2; \text{ and}$$

$$\text{Circle 1 -}\quad (x+\rho)^2 + y^2 = \rho^2 .$$

Solving for the intersection of Circle 1 with Lines 1 and 2 gives:

$$y_1 = (-\cos\alpha)\,(\rho\sin\alpha - w/2) + \sin\alpha\,\sqrt{\rho^2 - (\rho\sin\alpha - w/2)^2}\,,$$

$$x_1 = (\cos\alpha/\sin\alpha)\,y_1 - w/2\,\sin\alpha\,,$$

$$y_2 = (-\cos\alpha)\,(\rho\sin\alpha + w/2) + \sin\alpha\,\sqrt{\rho^2 - (\rho\sin\alpha + w/2)^2}\,,$$

$$x_2 = (\cos\alpha/\sin\alpha)\,y_2 + w/2\,\sin\alpha .$$

The angles covered are:

$$\Delta\varphi_1 = \tan^{-1}\left[y_1 \Big/ \left(\rho + x_1\right)\right], \text{ and}$$

$$\Delta\varphi_2 = \tan^{-1}\left[\left|y_2\right| \Big/ \left(\rho + x_2\right)\right] .$$

Then,

$$\begin{aligned}\text{COVERAGE/CIRCUIT} &= 2(\Delta\varphi_1 + \Delta\varphi_2)/\varphi\,,\\ &= 2\,(\Delta\varphi_1 + \Delta\varphi_2)/\left|k_n/f_n\right|\,2\pi\,,\\ &= \left|f_n/k_n\right|\,(\Delta\varphi_1 + \Delta\varphi_2)/\pi\,,\end{aligned}$$

where:

r = radius of surface at the parallel,

$\rho = \left| f_n / k_n \right| r$,

$\varphi = \left| k_n / f_n \right| \ (2\pi)$, and

$\alpha =$ helix angle at the parallel.

Case C-3 — Parallel in a Conical Section, Band Crosses the Parallel Once - For parallels near the turnaround parallel, the band will only cross the parallel one time (Line 2 of Figure C-3 does not intersect Circle 1). Figure C-4 shows this case. Again, the portion of the developed cone angle covered is computed.

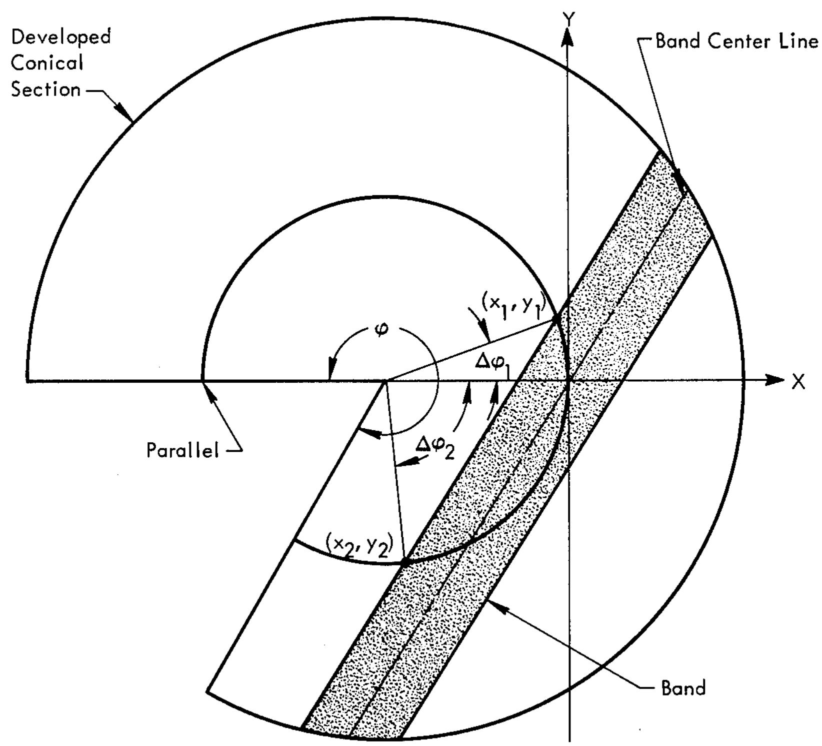

Figure C-4. DEVELOPED CONICAL SECTION. (Band Crosses the Parallel Once)

The points of intersection are:

$$y_1 = (-\cos\alpha)(\rho \sin\alpha - w/2) + \sin\alpha \sqrt{\rho^2 - (\rho\sin\alpha - w/2)^2} \ ,$$

$$x_1 = (\cos\alpha/\sin\alpha) y_1 - w/2 \sin\alpha \ ,$$

$$y_2 = (-\cos\alpha)(\rho \sin\alpha - w/2) - \sin\alpha \sqrt{\rho^2 - (\rho\sin\alpha - w/2)^2} \ ,$$

$$x_2 = (\cos\alpha/\sin\alpha) y_2 - w/2 \sin\alpha \ .$$

The angles covered are:

$$\Delta\varphi_1 = \tan^{-1}\left[y_1 / \left(\rho + x_1\right)\right],$$

$$\Delta\varphi_2 = \tan^{-1}\left[\left|y_2\right| / \left(\rho + x_2\right)\right] \text{ if } \rho > \left|x_2\right|,$$

$$= \tan^{-1}\left[\left|\rho + x_2\right| / y_2\right] + \pi/2 \text{ if } \left|x_2\right| \geq \rho .$$

Then,

$$\text{COVERAGE/CIRCUIT} = \left(\Delta\varphi_1 + \Delta\varphi_2\right)/\varphi ,$$

$$= \left(\Delta\varphi_1 + \Delta\varphi_2\right)/\left|k_n / f_n\right| 2\pi ,$$

$$= \left|f_n / k_n\right| \left(\Delta\varphi_1 + \Delta\varphi_2\right) / 2\pi .$$

Case C-4 — Parallel in a Conical Section; Parallel Beyond Turnaround Parallel – Since the band has a finite width, parallels beyond the turnaround parallel can be covered (turnaround parallel being that parallel where the geodesic turns around). This case is illustrated in Figure C-5. The portion of the angle covered is again computed.

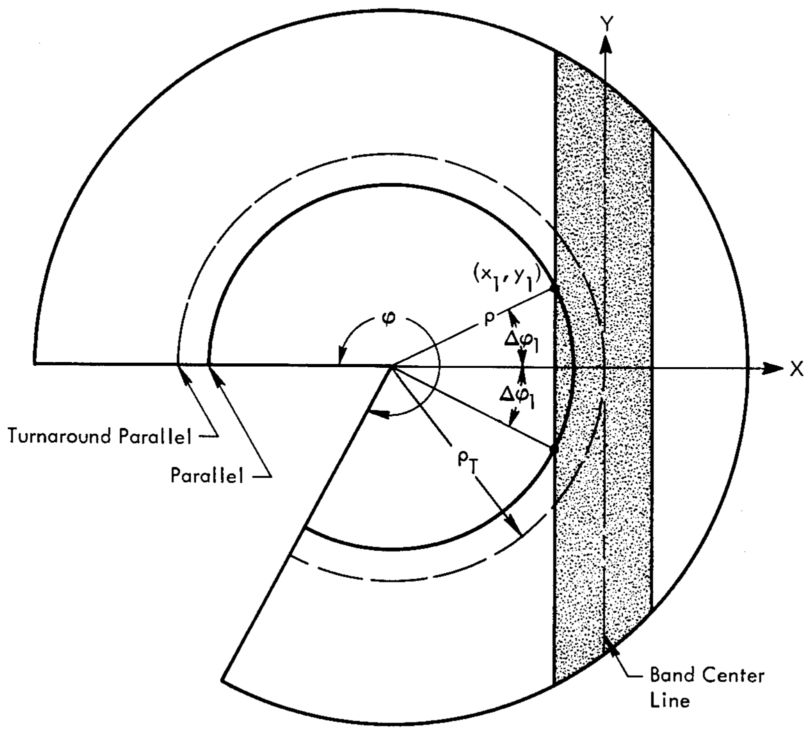

Figure C-5. PARALLEL BEYOND THE TURNAROUND PARALLEL.

Solving for the point of intersection gives:

$$y_1^2 + \left(\rho_T - w/2\right)^2 = \rho^2 \ ,$$

$$y_1 = \sqrt{\rho^2 - \left(\rho_T - w/2\right)^2} \ .$$

And the angle covered is:

$$\Delta\varphi_1 = \tan^{-1}\left[y_1 \Big/ \left(\rho_T - w/2\right)\right] \qquad \text{if } \rho > \rho_T - w/2$$

$$= 0 \ . \qquad \text{if } \rho \leq \rho_T - w/2$$

Then,

$$\text{COVERAGE/CIRCUIT} = 2\left(\Delta\varphi_1\right)/\varphi ,$$

$$= 2\left(\Delta\varphi_1\right)/\left|k_n / f_n\right| 2\pi ,$$

$$= \left|f_n / k_n\right| \Delta\varphi_1 / \pi \ .$$

APPENDIX D

COMPUTER PROGRAMS

Fortran Program

The geodesic program consists of two main programs and 17 subroutines. In addition, the plotting routines described here utilize several subroutines written for the Gerber Scientific Plotter.[6]

The subroutines are called by one of the main programs. Main program DESIGN is utilized when computing and plotting geodesic characteristics. Main program DEVPLT is used for plotting geodesics on a developed surface. The deck arrangements for the two operations are shown in Figures D-1 and D-2. Flow sheets of these two main programs are shown in Figures D-3 and D-4. The 17 subroutines are described in the sections that follow.

Subroutine PARMET - This routine computes various parameters for the conical and cylindrical sections that make up the surface. The parameters are stored and used by other routines called.

Subroutine DEVELP - This routine plots the developed surface. Certain parameters computed in this routine are utilized by the subroutine which plots a geodesic on the developed surface.

Subroutine DELTHA - This routine determines the delta theta (mandrel rotation) in each conical and cylindrical section and the total rotation for one circuit. The length of filament laid down in each section is also computed. The subroutine argument is the geodesic number.

Subroutine NOCIRC - This routine computes the number of circuits and the number of patterns necessary to lay down a given thickness at a desired parallel. The first argument of NOCIRC is the geodesic number. The second argument is a flag specifying which of two approaches should be used in determining the number of circuits per pattern. If the flag equals zero, the number of circuits per pattern will be such that one pattern will give the desired thickness of the given parallel. If the flag is one, the number of circuits per pattern will be determined so as to produce complete coverage at the specified parallel. The number of patterns necessary to build up the thickness will then be computed. Since the specified initial helix angle will not likely produce a geodesic having the desired number of circuits per pattern, the third argument of NOCIRC is a flag specifying which of two options should be taken in computing the geodesic. If the flag is zero, the initial helix angle is adjusted to produce a geodesic having the desired number of circuits per pattern. If the flag is one, the geodesic is distorted to obtain the number of circuits per pattern that is wanted.

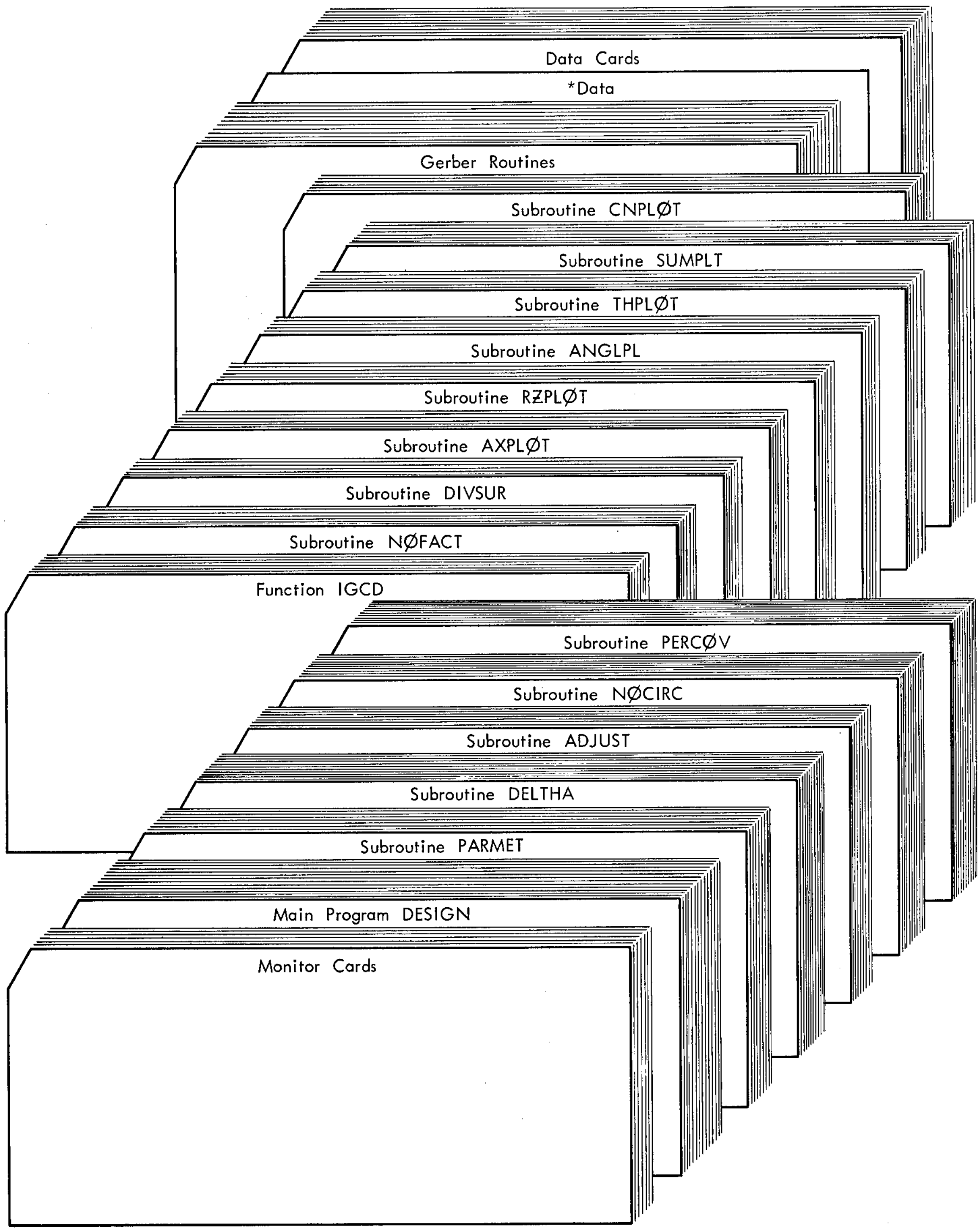

Figure D-1. CARD-DECK ARRANGEMENT FOR COMPUTING GEODESICS AND PLOTTING THEIR CHARACTERISTICS.

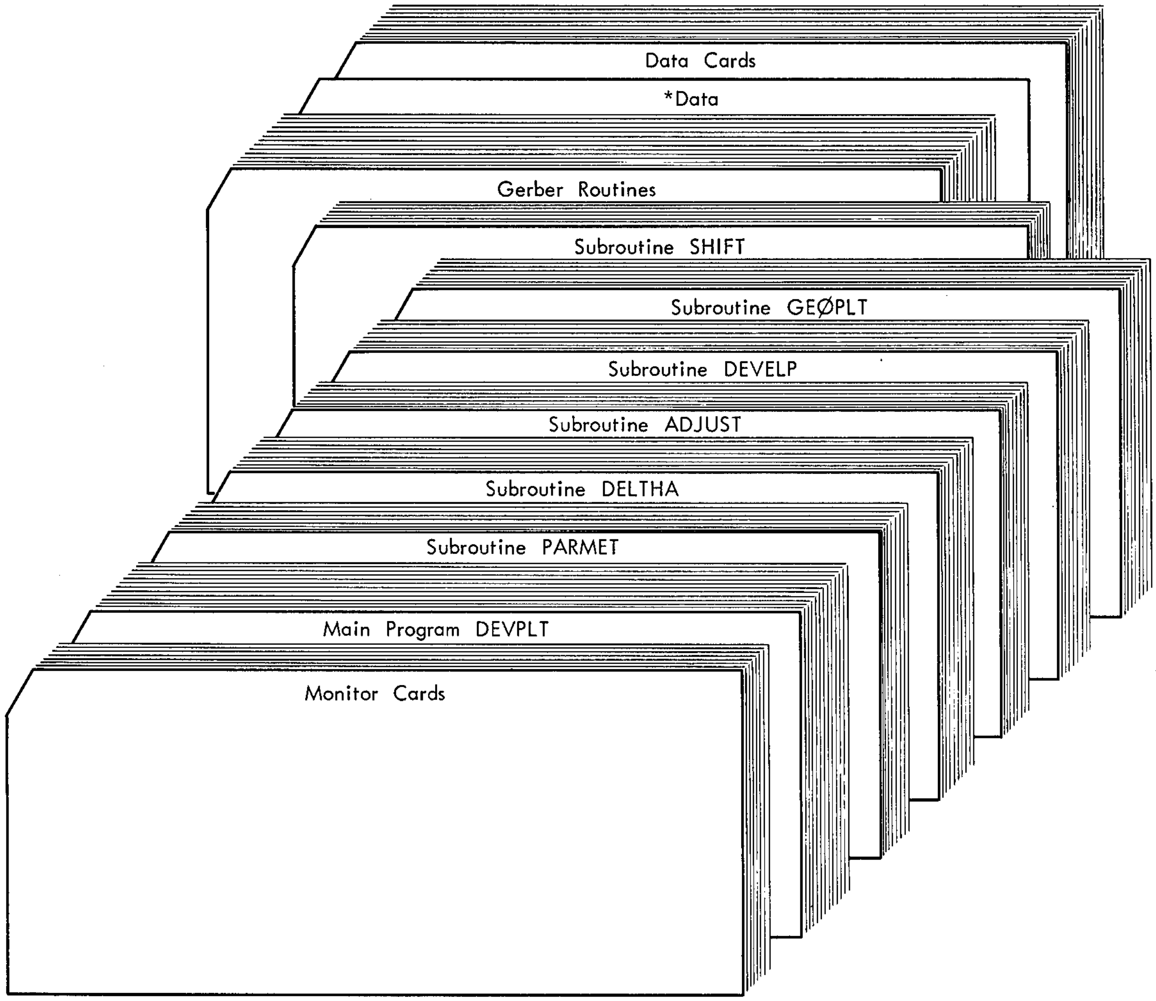

Figure D-2. CARD-DECK ARRANGEMENT FOR COMPUTING AND PLOTTING GEODESICS ON A DEVELOPED SURFACE.

Subroutine GEOPLT - This routine plots a geodesic on the developed surface. The arguments of GEOPLT are the geodesic number and the number of circuits to be plotted.

Subroutine DIVSUR - This routine computes at surface parallels, the helix angle at the parallel for each geodesic, the thickness produced by each geodesic, and the total thickness at the parallel. These values are written on magnetic tape for use by the plotting routines. The argument of DIVSUR is the interval along the contour at which the above described values will be computed.

Subroutine RZPLOT - This is the routine for plotting R and Z versus S (normalized). The first two arguments of the subroutine are the x and y coordinates of the origin for the plot. The third and fourth arguments are the lengths of the x and y axes, respectively.

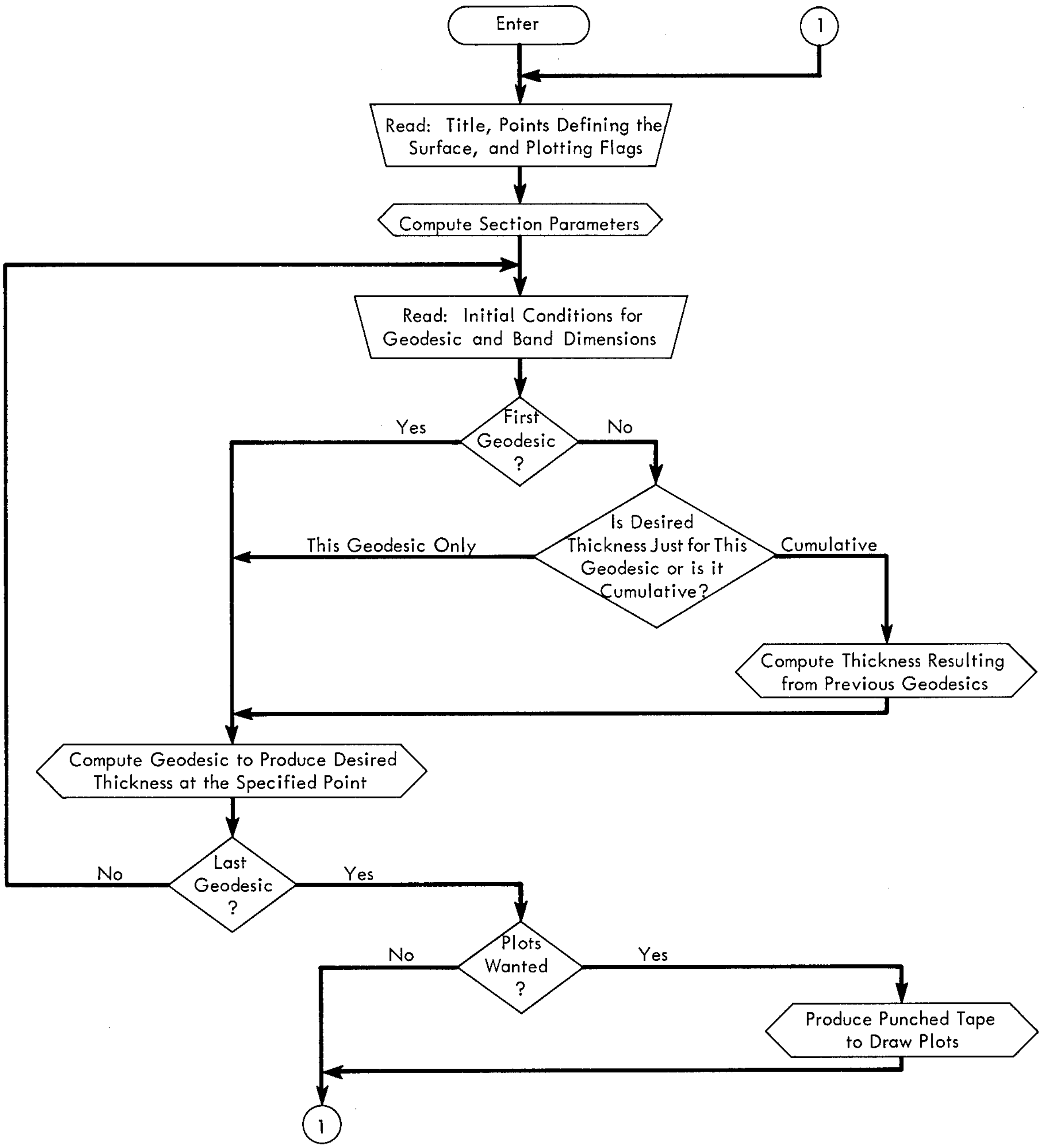

Figure D-3. FLOW SHEET OF THE MAIN PROGRAM "DESIGN".

Subroutine ANGLPL - This routine plots the helix angles versus S. The four arguments are the same as those of subroutine RZPLOT.

Subroutine THPLOT - This is the routine for plotting the thickness for a single geodesic versus S. This is a normalized plot with the thickness normalized with respect to the maximum thickness resulting from all of the geodesics. The first argument of THPLOT is the geodesic number. The next four arguments are the same as those of RZPLOT.

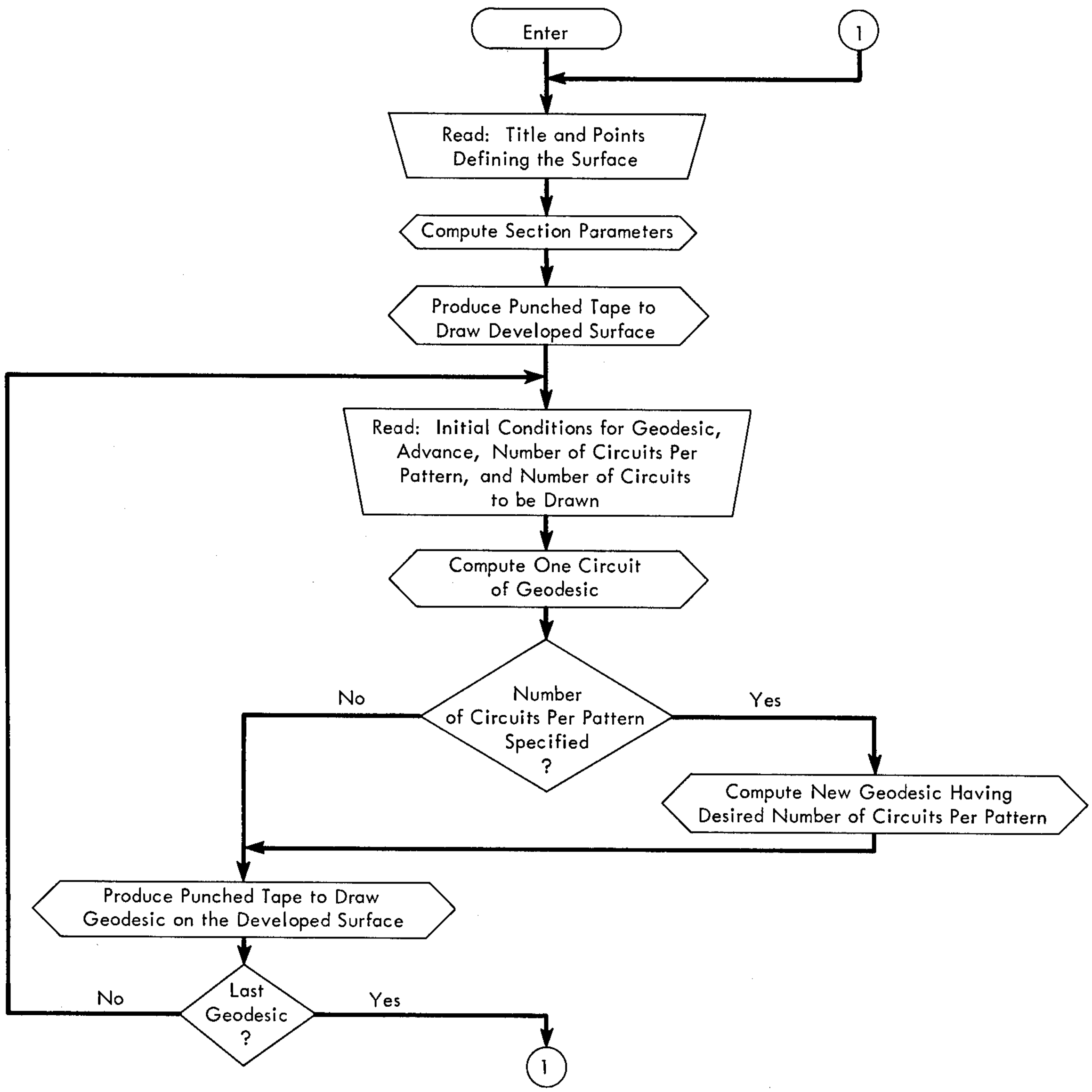

Figure D-4. FLOW SHEET OF THE MAIN PROGRAM "DEVPLT".

Subroutine SUMPLT - This routine plots the total thickness (normalized) resulting from all of the geodesics. The four arguments are the same as those of RZPLOT.

Subroutine CNPLOT - This is the routine for plotting the contour of the surface as it appears after the wrap. The first two arguments of CNPLOT are the origin for the plot. The third argument is the desired scale of the X (or Z) axis; the fourth is the scale of the Y (or R) axis.

<u>Subroutine PERCOV</u> - This routine is called by subroutines NOCIRC and DIVSUR to compute, at a given parallel, the coverage per circuit of the goedesic and the helix angle occurring at the parallel. The first argument of PERCOV is the radius of the part at the parallel. The next argument is the number of the section in which this parallel lies. The third argument is the geodesic number. The routine returns the coverage per circuit, which is the fourth argument, and the helix angle at the parallel, which is the fifth argument.

<u>Subroutine ADJUST</u> - This routine is called by NOCIRC to adjust the initial helix angle in order to obtain a geodesic having a predetermined number of revolutions per circuit. The first argument of ADJUST is the geodesic number. The second argument is the desired number of revolutions per pattern, and the third is the desired number of circuits per pattern. The fourth argument is the number of revolutions per circuit of the geodesic as initially specified. The fifth argument is the maximum difference that will be allowed between the number of revolutions per circuit of a new geodesic and the revolution per circuit desired. The sixth argument of ADJUST is a flag indicating to the calling program whether or not a geodesic could be found having the desired number of revolutions per circuit.

<u>Subroutine NOFACT</u> - This subroutine, called by NOCIRC, checks two integers for common factors. If the integers are found to have common factors, one (or) both is altered to obtain new integers having no common factors. The two arguments of NOFACT are the two integers involved.

<u>Function IGCD</u> - This is a Fortran function for determining the greatest common divisor of two integers. This function is called by subroutine NOFACT. The two arguments are the two integers whose greatest common divisor is desired.

<u>Subroutine SHIFT</u> - This routine is called by subroutine DEVELP if, when plotting the developed surface, two of the sections overlap. The first argument of SHIFT is the section number. The second argument, computed by the subroutine, is the amount of shift necessary to prevent the section from overlapping.

<u>Subroutine AXPLOT</u> - This routine, called by the various routines for plotting geodesic characteristics, is an axis generator. Its purpose is to draw and label the axes for a plot. The first two arguments are the x and y coordinates of the origin for the plot. The third and fourth arguments are the length of x and y axes, respectively. The fifth and sixth arguments are the divisions per inch, on the x axis and y axis, to be marked. The seventh and eighth arguments are the length each division represents (x and y axes). The ninth and tenth arguments specify which divisions are to be labeled; ie, if this number is one, every division will be labeled; if two, every other division will be labeled, etc. The eleventh argument is a flag indicating the size of letters to be used in labeling the axes. The twelveth and thirteenth arguments are the names of the x and y axes, respectively.

Input Format for the Main Program DESIGN

The input format for computing a geodesic and plotting its characteristics is shown in Figure D-5. The first card of the input is a title card containing alphanumeric information. The next card states the number of points to be used in defining the contour of the surface. The points defining the surface (r_n, z_n) are then given in order of increasing z, where z is the axis of revolution of the surface. The next card contains the plotting flags, one flag for each type plot available. Also included on this card is the interval along the surface contour (step length) at which thickness and helix angle are to be computed and plotted. (If all of the plotting flags are zero, thickness and helix angle will not be computed at all.) The scale of the final contour plot (1.0 = full scale, .5 = half scale, etc) is also included on this card. The next card indicates the number of geodesics to be wrapped. Then finally there is one card specifying each geodesic. This card contains the initial point of the geodesic (r_0 and/or z_0 is needed), the initial helix angle (this could be 90° if the initial point is the turnaround point), the desired thickness at a specified point (r and/or z needed), the band dimensions, and three flags. The first flag indicates which option is to be used in determining the number of circuits per pattern (Options 1 and 2). The second flag indicates whether a new geodesic having the desired circuits per pattern is to be found (adjust) or if the computed rotation of the geodesic is to be linearly distorted (distort) to produce the desired number of circuits per pattern. The final flag indicates whether the desired thickness is to be produced by the current geodesic alone or if it is the cumulative thickness of this and all prior geodesics.

Input Format for the Main Program DEVPLT

The input format for computing geodesics and plotting them on a developed surface is shown in Figure D-6. The format for the surface definition is identical to that used in main program DESIGN. One card is needed to specify each geodesic to be drawn. This card contains the initial radius (r_0), the initial helix angle (α_0), the advance per pattern, the number of circuits per pattern, and the number of circuits to be drawn. If the number of circuits per pattern is specified, a new geodesic will be computed (if necessary) to obtain one having the desired number of circuits per pattern. If this field is left blank, the geodesic as specified will be plotted.

Output of the Main Program DESIGN

Output of the main program DESIGN and its subroutines consists of the following:

(1) The alphanumeric information on the title card;
(2) The initial conditions for the geodesic;

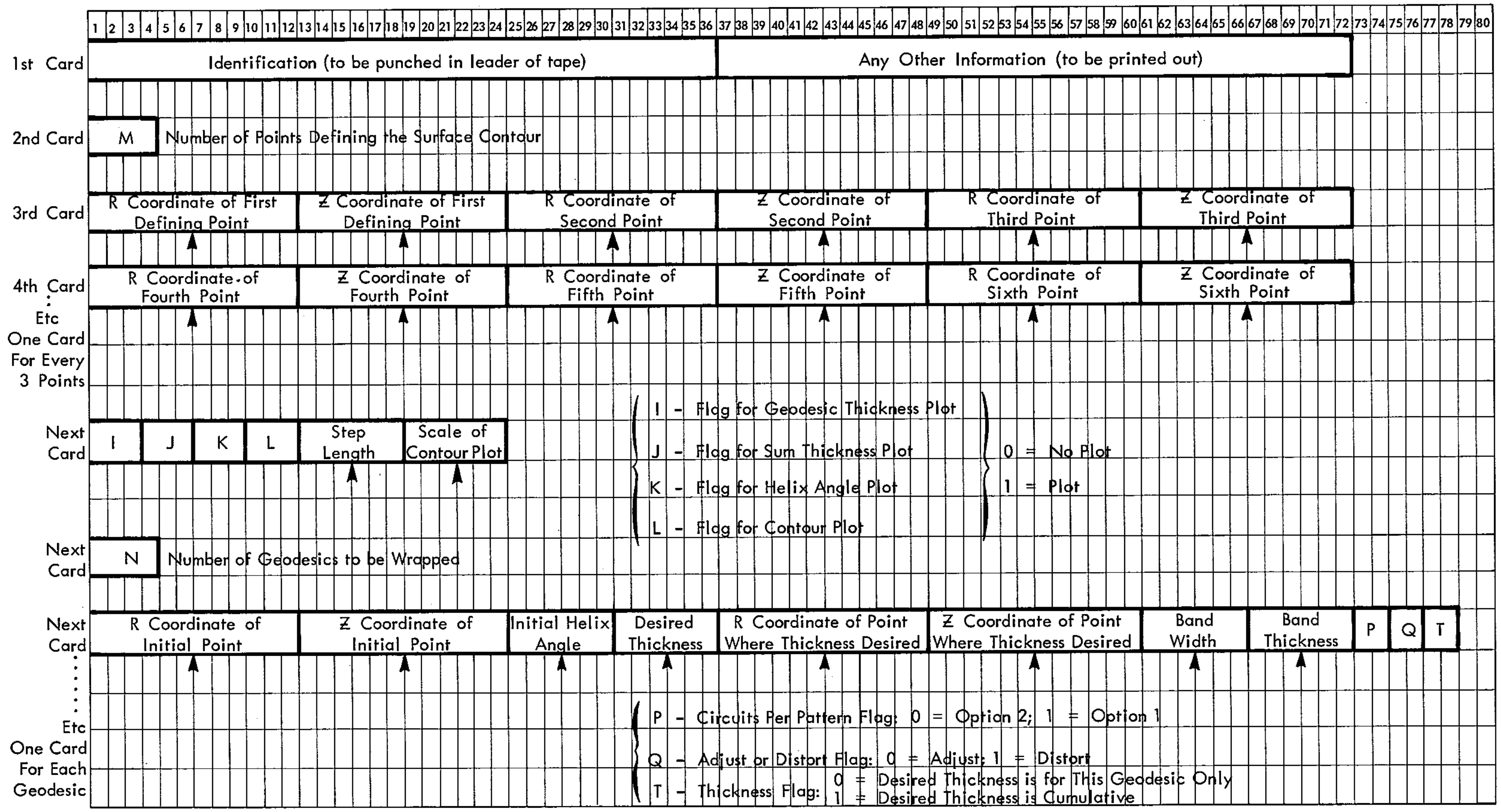

Figure D-5. INPUT FORMAT FOR THE MAIN PROGRAM "DESIGN".

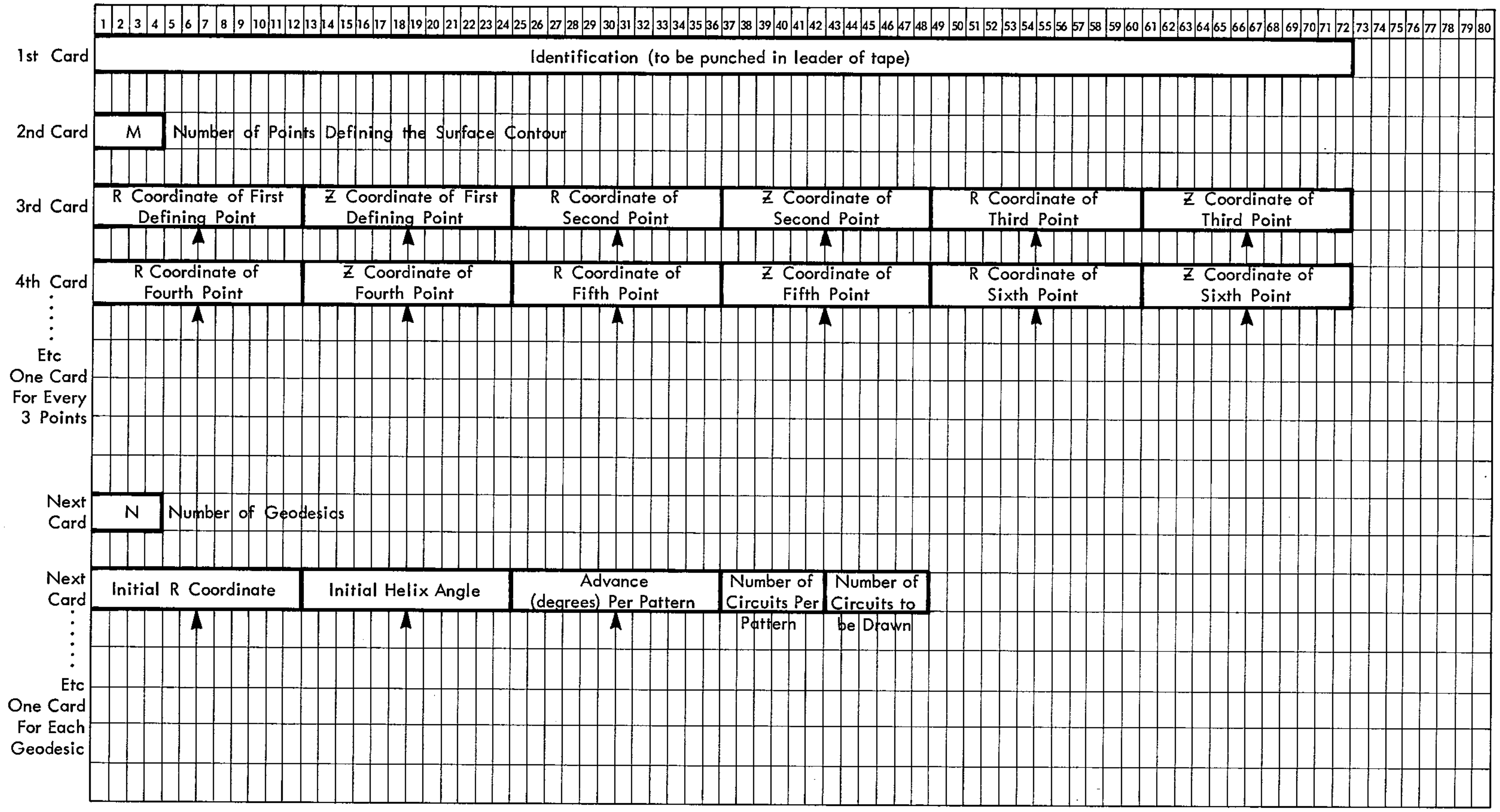

Figure D-6. INPUT FORMAT FOR THE MAIN PROGRAM "DEVPLT".

(3) Computed data for the geodesic which includes the geodesic number, the number of circuits necessary to produce the desired thickness, the number of patterns needed, the number of circuits per pattern, a ratio of integers which is the ratio of revolutions per pattern to circuits per pattern (ie, the number of revolutions per circuit), the computed thickness at the specified point resulting from this geodesic, and the parallels at which turnaround occurs;
(4) The distortion factor (the computed rotation is multiplied by this factor to achieve a wrap having the desired number of revolutions per circuit);
(5) The rotation (delta theta) and length of filament to be laid down in each conical and cylindrical section during one-half circuit; and
(6) The total rotation (degrees) and total length of filament for one circuit.

Output of the Main Program DEVPLT

Output of the main program DEVPLT and its subroutines consists of the following:

(1) The alphanumeric information on the title card;
(2) The initial conditions for the geodesic;
(3) The rotation and filament length for one circuit;
(4) A ratio of integers which is the ratio of revolution per pattern to circuits per pattern;
(5) The parallels at which turnaround occurs; and
(6) The rotation and filament length in each section for one-half circuit.

APT Program

The APT program represents the initial efforts on this project. Due to the limited amount of storage available in APT, this approach was abandoned and the Fortran program previously described was undertaken. Therefore, the APT program is limited to computing a geodesic and plotting it on the developed surface. The input to the program is a point definition of the contour to be wrapped. The 19 reserved words should be dimensioned at least as large as the number of points defining the contour.

MAC1 — This routine, utilizing the point definition of the surface contour, defines the developed surface and computes various section parameters. The argument of MAC1, M, is the number of points defining the contour.

MAC2 — This is the routine for plotting the developed surface. The argument, M, is again the number of points defining the contour.

MAC3 — This routine computes, for an initial helix angle and radius, one circuit of the geodesic. The initial helix angle is adjusted (if necessary) to produce a geodesic having a specified number of circuits per pattern. The first argument of

MAC3, RO, is the radius of the initial point. The second argument, AZERO, is the initial helix angle at the starting point. The third argument, PRIME, is the desired number of circuits per pattern and should be a prime number. If other than a prime number is specified, the program could determine a geodesic having fewer circuits per pattern than desired. The fourth argument, M, is the number of points defining the surface contour. The final argument, EPS, is the maximum allowable difference between the number of revolutions per circuit of the computed geodesic and the revolutions per circuit desired. This value will normally be small. However, if the user does not want the initial helix angle altered, a large value (say, EPS = 1) should be used.

MAC4 — This is the routine for plotting the geodesic on the developed surface. The first argument, TZERO, is the starting value of theta, θ_0. The second argument, J, is the section in which the plot will originate. The third argument, NUMBER, is the number of circuits to be drawn. The plot will begin at the left side of section J, proceed to the right, and terminate at the right hand side of section J-1. If the value specified for NUMBER is the same as that specified for PRIME in MAC3, the geodesic will return to its starting point (ie, complete one pattern).

A limited amount of program output appears after subroutine MAC3 has been executed. The parameters printed out are:

(1) Pass - The number of iterations of the initial helix angle to get desired number of circuits per pattern;

(2) Del - The difference between the desired number of revolutions per circuit and revolutions per circuit actually obtained;

(3) Rvn - revolutions per circuit obtained with adjusted initial helix angle;

(4) Integer - the integral part of Rvn;

(5) Fract - the fractional part of Rvn;

(6) Partn - the fractional part of desired number of revolution/circuit;

(7) N - the numerator of (N/PRIME) which results in the value of Partn;

(8) Alpha - the adjusted initial helix angle;

(9) Tsum - the rotation for one circuit in degrees;

(10) Cons - the constant of integration, $r_0 \sin \alpha_0$ (also radius at turnaround);

(11) L - the upper turnaround section;

(12) l - The lower turnaround section;

(13) Dtheta(n) - the rotation occurring on section n in one-half circuit (in degrees);

(14) Dbeta(n) - the angle traversed on the developed surface of section n (one-half circuit); and

(15) Flngth(n) - the length of filament on section n (one-half circuit).

Program Listing

The two main programs and the 17 subroutines previously described are listed below. Following this Fortran list is a listing the the APT macros.

```
*LABEL


CDESIGN    MAIN DESIGN PROGRAM FOR WINDING GEODESICS
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      COMMON AA, BB, CC, DEL, DELRHO, NSTART
      COMMON LLL , RHOMIN, FR , TMIN
      PI # 3.14159265
      REWIND 8
    5 READ INPUT TAPE 5 , 1000 ,  ( TITLE(K) , K # 1, 12 )
 1000 FORMAT  ( 12A6 )
      READ INPUT TAPE 5 , 1010 , M
 1010 FORMAT ( I4 )
      READ INPUT TAPE 5 , 1020 , ( R(N) , Z(N), N # 1 , M )
 1020 FORMAT ( 6F12.6 )
      READ INPUT TAPE 5 , 1030 , ITHICK , ISUM, IANGLE, ICON,STEP, SCLE
 1030 FORMAT ( 4I3 , 2F6.3 )
      CALL PARMET
      READ INPUT TAPE 5 , 1010 , NOGEOD
      DO  100   I # 1 , NOGEOD
      READ INPUT TAPE 5 , 1050 , RO(I), ZO(I),ALPHAO(I),THICK(I), RT(I),
     1 ZT(I), W(I) , D(I) , KK , LA , KTHICK
 1050 FORMAT ( 2F12.6 , 2F6.3 , 2F12.6 , 2F6.3 , 3I2  )
      WRITE OUTPUT TAPE 6 , 1080 , ( TITLE(K) , K # 1 ,12 )
 1080 FORMAT ( 1H1 , 12A6 )
      WRITE OUTPUT TAPE 6 , 1090
 1090 FORMAT ( 111H0    INITIAL    INITIAL    HELIX     DESIRED      AT PO
     1INT          BAND          BAND     CIRC/PAT  ADJUST THICKNESS  /
     2 108H       R           Z          ANGLE     THICKNESS    R          Z
     3      WIDTH    THICKNESS     FLAG     FLAG      FLAG  )
      WRITE OUTPUT TAPE 6 , 1100 , RO(I) , ZO(I) , ALPHAO(I) , THICK(I),
     1 RT(I) , ZT(I) , W(I) , D(I) , KK , LA , KTHICK
 1100 FORMAT ( 1H0 , 8F10.4 , 3I8 )
      IF ( I - 1 )    95 , 95 , 10
   10 IF ( KTHICK )   95 , 95 , 15
   15 IF ( ZT(I) )    20 , 20 , 70
   20 IF ( RT(I) )    25 , 25 , 50
   25 WRITE OUTPUT TAPE 6 , 2000 , I
 2000 FORMAT (31H0(RT,ZT) NOT GIVEN FOR GEODESIC , I3,13H (RO,ZO) USED )
      IF ( ZO(I) )    30 , 30 , 40
   30 IF ( RO(I) )    35 , 35 , 45
   35 WRITE OUTPUT TAPE 6 , 2010 , I
 2010 FORMAT ( 44H0STARTING STATION NOT SPECIFIED FOR GEODESIC , I3 ,
     1 16H CANNOT COMPUTE )
      GO TO  100
   40 ZT(I) # ZO(I)
      RT(I) # RO(I)
      GO TO  70
   45 RT(I) # RO(I)
   50 CONTINUE
```

```
      DO  55  N # 2 , M
      IF ( RT(I) - R(N) )  65 , 60 , 55
   55 CONTINUE
      WRITE OUTPUT TAPE 6 , 2020 , I
 2020 FORMAT ( 38HOCOULD NOT LOCATE (RT,ZT) FOR GEODESIC , I3 , 17H THIS
     1 ONE SKIPPED )
      GO TO 100
   60 N # N + 1
   65 NTH # N - 1
      GO TO  84
   70 CONTINUE
      DO  72  N # 2 , M
      IF ( ZT(I) - Z(N) )  78 , 76 , 72
   72 CONTINUE
      IF ( RT(I) )   74 , 74 , 50
   74 WRITE OUTPUT TAPE 6 , 2020 , I
      GO TO  100
   76 N # N + 1
   78 NTH # N - 1
      IF ( RT(I) )   82 , 82 , 84
   82 RT(I) # AK(NTH) * ( ZT(I) - Z(NTH) ) + R(NTH)
   84 CONTINUE
      IM1 # I - 1
      DO  88  K # 1 , IM1
      CALL  PERCOV ( RT(I) , NTH , K , PERCNT , HANGL )
      TK #  FLOATF ( NC(K) ) * D(K) * PERCNT
      THICK(I) # THICK(I) - TK
   88 CONTINUE
      IF ( THICK(I) )  90 , 90 , 95
   90 NC(I) # 0
      WRITE OUTPUT TAPE 6 , 2030 , I
 2030 FORMAT ( 13HOFOR GEODESIC , I3 , 87H THICKNESS BUILT UP BY PREVIOU
     1S LAYERS EXCEEDS DESIRED THICKNESS - THIS ONE NOT NEEDED )
      GO TO 100
   95 CONTINUE
      CALL  NOCIRC ( I, KK , LA )
  100 CONTINUE
      IF ( ITHICK + ISUM + IANGLE + ICON )  250  , 250 , 120
  120 IF ( STEP )   130 , 130 , 140
  130 STEP # .050
  140 CALL  DIVSUR ( STEP )
      XO # 0.0
      YO # 0.0
      IF ( ITHICK + ISUM + IANGLE )  210 , 210 , 142
  142 CONTINUE
      XL # 5.0
      YL # 5.0
      CALL  RZPLOT ( XO, YO, XL , YL )
      IF ( IANGLE )  160 , 160, 150
  150 CALL   ANGLPL ( XO, YO, XL , YL )
  160 CONTINUE
      IF ( ITHICK )  190 , 190  , 170
  170 DO   180   I # 1 , NOGEOD
  180 CALL  THPLOT ( I, XO, YO, XL, YL )
  190 CONTINUE
```

```
      IF  ( ISUM )  210  , 210  , 200
  200 CALL  SUMPLT ( XO, YO, XL , YL )
  210 CONTINUE
      IF ( ICON )  250  , 250 , 220
  220 IF ( SCLE )   230 ,  230 , 240
  230 SCLE # .5
  240 CONTINUE
      CALL  CNPLOT ( XO, YO, SCLE, SCLE )
  250 CONTINUE
      END FILE  8
      GO TO  5
      END
*LABEL


CDEVPLT      MAIN PROGRAM FOR PLOTTING GEODESIC ON DEVELOPED SURFACE
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      COMMON AA, BB, CC, DEL, DELRHO, NSTART
      COMMON LLL , RHOMIN, FR , TMIN
      PI # 3.14159265
      REWIND  8
   10 READ INPUT TAPE 5 , 1000 , ( TITLE(K) , K # 1, 12 )
 1000 FORMAT ( 12A6 )
      READ INPUT TAPE 5 , 1010 , M
 1010 FORMAT ( I4 )
      READ INPUT TAPE 5 , 1020 , ( R(N) , Z(N) , N # 1 , M )
 1020 FORMAT ( 6F12.6 )
      READ INPUT TAPE 5 , 1010 , NOGEOD
      CALL  PARMET
      CALL  DEVELP
      DO  200  I # 1 , NOGEOD
      READ INPUT TAPE 5 , 1030 , RO(I) , ALPHAO(I), ADVDEG, NCPERP, NUM
 1030 FORMAT ( 3F12.6 , 2I6 )
      DO  20  N # 2 , M
      IF ( RO(I) - R(N) )   40 , 30 , 20
   20 CONTINUE
      WRITE OUTPUT TAPE 6 , 2000 , I
 2000 FORMAT ( 52H0 COULD NOT DETERMINE STARTING SECTION FOR GEODESIC ,
     1 I3 , 20H , THIS ONE SKIPPED   )
      GO TO  200
   30 N # N + 1
   40 NSTART # N - 1
      IF ( AK(NSTART)) 50 , 60 , 50
   50 ZO(I) # ( RO(I) - R(NSTART) ) / AK(NSTART) + Z(NSTART)
      GO TO  70
   60 ZO(I) # Z(NSTART)
   70 CONTINUE
      CALL  DELTHA(I)
      NLOW # NLOW
      NHIGH # NHIGH
```

```
      IF ( NCPERP )  145 , 145 , 80
   80 RVN # TSUM / 360.0
      INTGR # RVN
      FRACT # RVN - FLOATF (INTGR)
      NB # NCPERP
      ANB # NB
      ADVNCE # ADVDEG / ( ANB * 360.0 )
      N # 1
      AN # 1.0
      PARTN # 1.0 / ANB
   90 AN # AN + 1.0
      N # N + 1
      PARTN1 # PARTN
      PARTN # AN / ANB
      IF ( FRACT - PARTN )   110 , 150 , 100
  100 IF ( N - NB + 1 )  90 , 130 , 130
  110 IF ( ABSF( FRACT-PARTN) - ABSF( FRACT-PARTN1) ) 130, 130 , 120
  120 N # N - 1
      PARTN # PARTN1
  130 CONTINUE
      NA # N
      EPS # .000001
      NAA # NB * INTGR + NA
      CALL  ADJUST ( I , NAA , NB , RVN , EPS , LL )
      NLOW # NLOW
      NHIGH # NHIGH
      IF ( LL )  150 , 150 , 140
  140 WRITE OUTPUT TAPE 6 , 2010 , I , ALPHAO(I) , RO(I)
 2010 FORMAT (  28H0 COULD NOT ADJUST GEODESIC , I3 , 53H , THEREFORE PL
     1OT IS FOR GEODESIC HAVING HELIX ANGLE , F6.3, 11H AT RADIUS ,F6.3)
  145 NA # 0
      NB # 0
      INTGR # 0
      ADVDEG # 0.0
  150 CONTINUE
      ZLOW # ( CONS(I) - R(NLOW) )  / AK(NLOW)  + Z(NLOW)
      ZHIGH # ( CONS(I) - R(NHIGH ) ) / AK(NHIGH) + Z(NHIGH)
      WRITE OUTPUT TAPE 6 , 2020 , ( TITLE(K) , K # 1 , 12 )
 2020 FORMAT ( 1H1 , 12A6 )
      WRITE OUTPUT TAPE 6 , 2030
 2030 FORMAT (  93H0GEODESIC HELIX      AT     ADVANCE    TOTAL    FILAM
     1ENT   INTEGERS       TURNAROUND STATIONS  /
     1 99H  NUMBER  ANGLE     RADIUS    PER PAT   ROTATION   LENGTH    N +
     3A / B     RADIUS    Z LOWER    Z UPPER    )
      WRITE OUTPUT TAPE 6 , 2040 , I, ALPHAO(I) , RO(I), ADVDEG , TSUM ,
     1 FLSUM , INTGR , NA , NB , CONS(I) , ZLOW , ZHIGH
 2040 FORMAT ( 1H0 , I4 , 5F10.3 , 3I4 , 3F10.3 )
      WRITE OUTPUT TAPE 6,2050, ( N,DTHETA(N),FLNGTH(N),N#NLOW,NHIGH )
 2050 FORMAT ( 1H0 / 39H0 SECTION  DELTA THETA  FILAMENT LENGTH  /
     1 ( 1H , I4 , 2F16.6 )  )
      WRITE OUTPUT TAPE 6 , 2060 , TSUM , FLSUM
 2060 FORMAT ( 8H0CIRCUIT  , F13.6 , F16.6  )
      IF ( NUM )  160 , 160 , 170
  160 NUM # 1
  170 CONTINUE
```

```
      CALL  GEOPLT ( I , NUM )
  200 CONTINUE
      GO TO 10
      END
*LABEL
CPARMET                   COMPUTE SECTION PARAMETERS
      SUBROUTINE   PARMET
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      XI(1) # 0.0
      MM # M - 1
      RMAX # R(1)
      DO  1200  N # 1 , MM
      IF ( ABSF( Z(N) - Z(N+1)) - .0001 )    1010 , 1010 , 1040
 1010 IF ( R(N+1) - R(N) )  1020 , 1020 , 1030
 1020 AK(N) #  -  ( 1.0E 20 )
      GO TO 1035
 1030 AK(N) #  1.0 E 20
 1035 F(N) #  1.0 E 20
      XI(N+1) # XI(N) + ABSF( R(N+1) - R(N) )
      GO TO 1200
 1040 AK(N) #  ( R(N+1) - R(N) ) / ( Z(N+1) - Z(N) )
      IF ( ABSF( AK(N) ) - .0001 )  1050 , 1050 , 1100
 1050 AK(N) # 0.0
 1100 F(N) # SQRTF( 1.0 +  AK(N)**2 )
      XI(N+1) #  XI(N) +  ( Z(N+1) - Z(N) ) * F(N)
 1200 RMAX # MAXIF ( R(N+1) , RMAX )
      ZMAX # Z(M)
      RETURN
      END
*LABEL


CDEVELP
      SUBROUTINE  DEVELP
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      DIMENSION R1(100),R2(100),PHI(100),XC(100)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      COMMON  AA,BB,CC,DEL,DELRHO,NSTART
      COMMON  LLL , RHOMIN , FR , TMIN
      COMMON  R1 , R2 , PHI , XC
      SHIFT1 # 0.0
      M1 # M - 1
      DO  200  N # 1 , M1
      IF ( AK(N) )  10 , 40 , 70
   10 FOK # - F(N) / AK(N)
      R1(N) # R(N+1) * FOK
```

```
      R2(N) # R(N) * FOK
      PHI(N) # 360.0 / FOK
      XC(N) # X1(N) + R2(N)
      IF ( AK(N) - AK(N-1) - .000001 )  30 , 30 , 20
   20 CALL SHIFT ( N , SHIFT2 )
      SHIFT1 # SHIFT1 + SHIFT2
   30 XC(N) # XC(N) + SHIFT1
      GO TO 200
   40 R2(N) # 2.0 * PI * R(N)
      IF ( AK(N-1) )  50 , 60 , 60
   50 CALL SHIFT ( N , SHIFT2 )
      SHIFT1 # SHIFT1 + SHIFT2
   60 XC(N) # X1(N) + SHIFT1
      R1(N) # X1(N+1) + SHIFT1
      GO TO 200
   70 FOK # F(N) / AK(N)
      R1(N) # R(N) * FOK
      R2(N) # R(N+1) * FOK
      PHI(N) # 360.0 / FOK
      XC(N) # X1(N) - R1(N)
      IF ( N - 1 )  100 , 100 , 80
   80 IF ( AK(N) - AK(N-1) - .000001 )  100 , 100 , 90
   90 CALL SHIFT ( N , SHIFT2 )
      SHIFT1 # SHIFT1 + SHIFT2
  100 XC(N) # XC(N) + SHIFT1
  200 CONTINUE
B     TITLE(7) # 242565254346
B     TITLE(8) # 472524606264
B     TITLE(9) # 512621232560
B     TITLE(10)# 474346635360
      CALL SETUP ( TITLE )
      CALL PLOT ( 0.0 , 0.0 , 1 , 2  )
      DO  300  N # 1 , M1
      IF ( AK(N) )  220 , 250 , 260
  220 CALL CIRCLE ( XC(N) , 0.0 , R2(N) , 180.0 , - PHI(N) , -1 )
      IF ( R1(N) - .000001 )  240 , 240 , 230
  230 CALL CIRCLE ( XC(N) , 0.0 , R1(N) , 180.0 - PHI(N) , PHI(N) , 1 )
  240 GO TO 300
  250 CALL  PLOT ( XC(N) , 0.0 , 1 , 2  )
      CALL  PLOT ( XC(N) , R2(N) , 1 , 1 )
      CALL  PLOT ( R1(N) , R2(N) , 1 , 1 )
      CALL  PLOT ( R1(N) , 0.0 , 1 , 1 )
      GO TO  300
  260 IF ( R1(N) - .000001 )   280 , 280 , 270
  270 CALL CIRCLE  ( XC(N) , 0.0 , R1(N) , 0.0 , PHI(N) , -1 )
  280 CALL CIRCLE  ( XC(N) , 0.0 , R2(N) , PHI(N) , - PHI(N) , 1 )
  300 CONTINUE
      CALL  PLOT  ( 0.0 , 0.0  , 1 , 1  )
      CALL  FINISH ( 30 , TITLE )
      END FILE 8
      RETURN
      END
*LABEL
```

```
CDELTHA            COMPUTE DELTA THETAS FOR GEODESIC I
      SUBROUTINE  DELTHA ( I )
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      CONV # 180.0 / PI
      CONS(I) # RO(I) * SINF ( ALPHAO(I) / CONV )
      DO  20  N # 2 , M
      IF ( ZO(I) - Z(N) )    40 , 30 , 20
   20 CONTINUE
   30 IF ( ALPHAO(I) - 90.0 )  38 , 33 , 38
   33 IF ( AK(N) )  40 , 40 , 38
   38 N # N + 1
   40 NSTART # N - 1
      IF ( CONS(I) - R(1) ) 150 ,  45  , 45
   45 J # NSTART + 1
   50 J # J - 1
      IF ( R(J) - CONS(I) )  60 , 60 , 50
   60 NLOW # J
      IF ( CONS(I) - R(M) ) 150 , 70 , 70
   70 J # NSTART
   80 J # J + 1
      IF ( R(J) - CONS(I) )   90 , 90 , 80
   90 NHIGH # J - 1
      FOK # F(NLOW) / AK(NLOW)
      NL1 # NLOW + 1
      TERM # ( R(NL1) / CONS(I) )**2 - 1.0
      IF ( TERM )  92 , 92 , 95
   92 ASEC2 # 0.0
      GO TO  98
   95 ASEC2 # ATANF ( SQRTF ( TERM ) )
   98 DBETA # ASEC2
      DTHETA(NLOW) # FOK * DBETA * CONV
      FLNGTH(NLOW) #  R(NL1) * FOK * SINF( DBETA )
      NH1 # NHIGH - 1
      IF ( NH1 - NL1 )  135 , 100 , 100
  100 DO 130  N # NL1 , NH1
      IF ( AK(N) )  120 , 110 , 120
  110 DBETA#CONS(I)*(Z(N+1)-Z(N)) / (R(N)*SQRTF(R(N)**2 - CONS(I)**2))
      DTHETA(N) # DBETA * CONV
      FLNGTH(N) # SQRTF(( Z(N+1)-Z(N))**2 + (R(N)*DBETA)**2 )
      GO TO  130
  120 FOK # ABSF( F(N) / AK(N) )
      ASEC1 # ASEC2
      TERM # ( R(N+1) / CONS(I) )**2 - 1.0
      IF ( TERM )  122 , 122 , 125
  122 ASEC2 # 0.0
      GO TO  126
  125 ASEC2 # ATANF ( SQRTF ( TERM ) )
  126 DBETA # ABSF ( ASEC2 - ASEC1 )
      DTHETA(N) # FOK * DBETA * CONV
      RN2 # R(N) * FOK
```

```
      RN3 # R(N+1) * FOK
      TERM # RN2**2 + RN3**2 - 2.0 * RN2 * RN3 * COSF ( DBETA )
      IF ( TERM )  127 , 127 , 129
  127 FLNGTH(N) # 0.0
      GO TO  130
  129 FLNGTH(N) # SQRTF ( TERM )
  130 CONTINUE
  135 CONTINUE
      FOK # ABSF ( F(NHIGH) / AK(NHIGH) )
      DBETA # ABSF ( ASEC2 )
      DTHETA(NHIGH) # FOK * DBETA * CONV
      FLNGTH(NHIGH) # FOK * R(NHIGH) * SINF (DBETA)
      TSUM # 0.0
      FLSUM # 0.0
      DO  140  N # NLOW , NHIGH
      TSUM # TSUM + DTHETA(N)
  140 FLSUM # FLSUM + FLNGTH(N)
      TSUM # 2.0 * TSUM
      FLSUM # 2.0 * FLSUM
      GO TO  160
  150 CONS(I) # MAXIF ( R(1) , R(M) )
      ALPHAO(I) # ATANF( CONS(I)/ SQRTF( RO(I)**2 - CONS(I)**2) ) * CONV
      WRITE OUTPUT TAPE 6 , 8000 , I , ALPHAO(I)
 8000 FORMAT ( 33H0 TURN-AROUND RADIUS FOR GEODESIC ,I3 , 56H  IS LESS T
     1HAN R(1) OR R(M) - STARTING ANGLE CHANGED TO , F10.6 )
      GO TO  45
  160 CONTINUE
      RETURN
      END
*LABEL


CNOCIRC       COMPUTE NUMBER OF CIRCUITS TO GIVE THICKNESS
      SUBROUTINE NOCIRC ( I , KK , LA )
C
C     I IS GEODESIC NUMBER
C     KK IS OPTION IN DETERMINING NUMBER OF CIRCUITS PER PATTERN
C     LA IS OPTION TO ADJUST STARTING ANGLE OR DISTORT GEODESIC
C
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      PGLASS # 1.0
      IF ( D(I) )    10 , 10 , 20
   10 D(I) # .001
      WRITE OUTPUT TAPE 6,1400 , I
 1400 FORMAT ( 35H0  DIAMETER OF ROVING FOR GEODESIC , I3 , 23H NOT GIVE
     1N - .001 USED   )
   20 IF ( W(I) )  30 , 30 , 40
   30 W(I) #   .1
      WRITE OUTPUT TAPE 6 ,1410 , I
 1410 FORMAT ( 32H0  WIDTH OF ROVING FOR GEODESIC , I3 ,  21H NOT GIVEN
```

```
     1- .1 USED  )
  40 IF ( THICK(I) )  42 , 42 , 48
  42 THICK(I) # 2.0 * D(I)
     WRITE OUTPUT TAPE 6,1415, I
1415 FORMAT(32H0 DESIRED THICKNESS FOR GEODESIC , I3, 29H NOT SPECIFIED
     1 , 2 D(I) USED )
  48 IF ( ZO(I) )   50 , 50 , 120
  50 IF ( RO(I) )   60 , 60 , 70
  60 WRITE OUTPUT TAPE 6,1420 , I
1420 FORMAT ( 46H0  STARTING STATION NOT SPECIFIED FOR GEODESIC , I3 ,
     1 16H CANNOT COMPUTE  )
     GO TO 500
  70 DO  80  N # 2 , M
     IF ( RO(I) - R(N) )   90 , 85 , 80
  80 CONTINUE
     WRITE OUTPUT TAPE 6,1430 , I
1430 FORMAT ( 72H0  WITH ZO NOT GIVEN , COULD NOT DETERMINE STARTING SE
     1CTION FOR GEODESIC , I3 , 10H USING RO  )
     GO TO  500
  85 N # N+1
  90 NSTART # N - 1
     IF ( AK(NSTART) )  100 , 110, 100
 100 ZO(I) # ( RO(I) - R(NSTART) ) / AK(NSTART)  + Z(NSTART)
     GO TO 160
 110 ZO(I) # Z(NSTART)
     GO TO  160
 120 DO  130  N #  2 , M
     IF ( ZO(I) - Z(N) )   140, 135,  130
 130 CONTINUE
     WRITE OUTPUT TAPE 6,1440,I
1440 FORMAT ( 63H0  USING ZO , COULD NOT DETERMINE STARTING SECTION FOR
     1 GEODESIC , I3 )
     GO TO  500
 135 N # N + 1
 140 NSTART # N - 1
     IF ( RO(I) )   150 , 150 , 160
 150 RO(I) # AK(NSTART) * ( ZO(I) - Z(NSTART) )  + R(NSTART)
 160 CONTINUE
     IF( ZT(I) )  170 , 170 , 240
 170 IF( RT(I) )  180 , 180 , 190
 180 WRITE OUTPUT TAPE 6 ,1450 , I
1450 FORMAT(31H0(RT,ZT) NOT GIVEN FOR GEODESIC , I3 ,13H (RO,ZO) USED )
 185 RT(I) # RO(I)
     ZT(I) # ZO(I)
     NTH # NSTART
     GO TO  280
 190 DO  200  N # 2 , M
     IF ( RT(I) - R(N) )  210, 205,  200
 200 CONTINUE
     WRITE OUTPUT TAPE 6,1460 , I
1460 FORMAT ( 83H0 WITH ZT NOT GIVEN , COULD NOT DETERMINE SECTION TO C
     1OMPUTE THICKNESS FOR GEODESIC , I3  , 39H USING RT , SO (RO,ZO) US
     2ED FOR (RT,ZT)  )
     GO TO  185
 205 N # N+1
```

```
210 NTH # N - 1
    IF ( AK(NTH) )  220 , 230 , 220
220 ZT(I) # ( RT(I) - R(NTH) ) / AK(NTH)  +  Z(NTH)
    GO TO  280
230 ZT(I) # Z(NTH)
    GO TO  280
240 DO  250  N # 2, M
    IF ( ZT(I) - Z(N) ) 260 , 255,  250
250 CONTINUE
    WRITE OUTPUT TAPE 6,1470 , I
1470 FORMAT ( 80H0 USING GIVEN ZT , COULD NOT DETERMINE SECTION TO COMP
   1UTE THICKNESS FOR GEODESIC,I3,28H SO (RO,ZO) USED FOR (RT,ZT)   )
    GO TO 185
255 N # N + 1
260 NTH # N - 1
    IF ( RT(I) )   270 , 270 , 280
270 RT(I) # AK(NTH) * ( ZT(I) - Z(NTH) ) + R(NTH)
280 CONTINUE
290 CONS(I) # RO(I) * SINF ( ALPHAO(I) * PI / 180.0 )
    CALL  PERCOV  ( RT(I) , NTH , I , PERCNT , HANGL )
    IF ( PERCNT )   300 , 300 , 310
300 RT(I) # CONS(I)
    ZT(I) # 0.0
    WRITE OUTPUT TAPE 6 ,1480, I
1480 FORMAT ( 42H0 COVERAGE AT (RT,ZT) IS ZERO FOR GEODESIC , I3 , 38H
   1,  TURNAROUND POINT USED FOR (RT,ZT)  )
    GO TO  160
310 IF ( KK )  320 , 320 , 330
320 B # THICK(I) / ( D(I) * PERCNT ) * PGLASS
    GO TO  340
330 B # 2.0 / PERCNT
340 NB # B + .5
    CALL DELTHA ( I )
    NLOW # NLOW
    NHIGH # NHIGH
    RVN # TSUM / 360.0
    INTGR # RVN
    FRACT # RVN - FLOATF( INTGR )
    A # B * FRACT
    NA # A + .5
    IF ( NA )   350 , 350 , 360
350 NA # 1
    GO TO  388
360 IF ( NA - NB )  380 , 370 , 375
370 NA # NB - 1
    GO TO  388
375 INTGR # INTGR + 1
    NA # NA - NB
380 CALL  NOFACT ( NA , NB )
388 CONTINUE
    EPS # .000001
    IF ( LA )  382 , 382 , 390
382 CONTINUE
    NAA # NB * INTGR + NA
    CALL  ADJUST ( I , NAA , NB , RVN , EPS , LL )
```

```
      IF ( LL )  384 , 384 , 390
 384  CALL  PERCOV  ( RT(I) , NTH , I , PERCNT , HANGL )
      DISTRT # 1.0
      GO TO  420
 390  RVN2 # FLOATF( INTGR )  +  FLOATF( NA ) / FLOATF ( NB )
      DISTRT # RVN2 / RVN
      DO  410  N # NLOW , NHIGH
 410  DTHETA(N) # DTHETA(N) * DISTRT
      TSUM # TSUM * DISTRT
 420  AN # THICK(I) / ( D(I) * PERCNT ) * PGLASS
      NOPATN    # AN / FLOATF( NB )  + .5
      NCPERP    # NB
      NC(I) # NCPERP  *   NOPATN
      THNESS # FLOATF ( NC(I) ) * D(I) * PERCNT / PGLASS
      ZLOW # ( CONS(I) - R(NLOW) ) / AK(NLOW)  + Z(NLOW)
      ZHIGH #( CONS(I) - R(NHIGH)) / AK(NHIGH) + Z(NHIGH)
      WRITE OUTPUT TAPE 6 ,1490 , I, NC(I) ,NOPATN   , NCPERP   , INTGR,
     1 NA, NB, THNESS , CONS(I) , ZLOW , ZHIGH
1490  FORMAT ( 100H0              NO. OF     NO. OF    CIRC. PER     RATIO OF
     1 INTEGERS     THICKNESS      TURNAROUND STATIONS / 106H GEODESIC  CI
     2RCUITS  PATTERNS  PATTERN        N    +     A    /     B  AT (RT,ZT) RA
     3DIUS     Z LOWER    Z UPPER   / 1H0 , I4, 6I10 , 4F10.6  )
      WRITE OUTPUT TAPE 6,1500, DISTRT
1500  FORMAT ( 23H0  DISTORTION FACTOR #   , F10.6 )
      WRITE OUTPUT TAPE 6,1510,(N,DTHETA(N), FLNGTH(N), N# NLOW,NHIGH )
1510  FORMAT ( 39H0SECTION  DELTA THETA  FILAMENT LENGTH   / ( 1H , I4 ,
     1 2F16.6  ) )
      WRITE OUTPUT TAPE 6 ,1520 , TSUM , FLSUM
1520  FORMAT ( 8H0CIRCUIT , F13.6 , F16.6 / 1H0   )
 500  CONTINUE
      RETURN
      END
*LABEL


CGEOPLT        PLOT GEODESIC ON DEVELOPED SURFACE
      SUBROUTINE  GEOPLT ( I , NUM )
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      DIMENSION R1(100 ) , R2(100 ) , PHI(100 ) , XC(100 )
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      COMMON  AA,BB,CC,DEL,DELRHO,NSTART
      COMMON  LLL , RHOMIN , FR , TMIN
      COMMON  R1 , R2 , PHI , XC
      KOUNT # 0
      CONV #  PI / 180.0
      THETA # 0.0
      WRITE OUTPUT TAPE 0,4010 , I
4010  FORMAT ( 28H DEVELOPED PLOT OF GEODESIC , I3 , 5H$       )
      READ INPUT TAPE 0,4020, ( TITLE(K) , K # 7,12 )
4020  FORMAT ( 6A6 )
      CALL  SETUP ( TITLE )
```

```
      DO 10  N # 1 , M
      IF ( RMAX - R(N) - .000001 )  20 , 20 , 10
   10 CONTINUE
   20 NSTART # N
      ZZ # Z(NSTART)
      UPDOWN # 1.0
   30 IF ( AK(N) )   40 , 420, 70
   40 AKK # -1.0
      IF ( UPDOWN )  50 , 50 , 60
   50 RR # R1(N)
      GO TO  100
   60 RR # R2(N)
      GO TO  100
   70 AKK #   1.0
      IF ( UPDOWN )   80 , 80 , 90
   80 RR # R2(N)
      GO TO  100
   90 RR # R1(N)
  100 AOF # ABSF ( AK(N) / F(N) )
      BETA # AOF * THETA
      XO # AKK * RR * COSF ( BETA * CONV )
      YO # RR * SINF ( BETA * CONV )
      IF ( AKK )  110 , 110 , 160
  110 IF (  N - NHIGH )   130 ,  170 , 170
  130 IF ( UPDOWN )  140 , 140 , 150
  140 RE # R2(N)
      GO TO  210
  150 RE # R1(N)
      GO TO  210
  160 IF ( N - NLOW )  170 , 170 , 180
  170 RE # RR
      BETAZ # BETA + 2.0 * AOF * DTHETA(N)
      GO TO  220
  180 IF ( UPDOWN )  190 , 190 , 200
  190 RE # R1(N)
      GO TO  210
  200 RE # R2(N)
  210 BETAZ # BETA + AOF * DTHETA(N)
  220 CONTINUE
      XD # AKK * RE * COSF ( BETAZ * CONV )
      YD # RE * SINF ( BETAZ * CONV )
      IF ( BETAZ - PHI(N) )   230 , 360 , 360
  230 XORE   # XO + XC(N)
      XDRE   # XD + XC(N)
      CALL  PLOT ( XORE  , YO , 1 , 2 )
      CALL  PLOT ( XDRE  , YD , 1 , 1 )
      THETA # BETAZ / AOF
      IF ( AKK )  240 , 290 , 290
  240 IF ( N - NHIGH )  250 , 270 , 270
  250 IF ( UPDOWN )  280 , 280, 260
  260 ZZ # Z(N+1)
      N # N + 1
      GO TO  340
  270 UPDOWN # - 1.0
  280 ZZ # Z(N)
```

```
      N # N - 1
      GO TO  340
  290 IF ( N - NLOW )  320 , 320 , 300
  300 IF ( UPDOWN )  310 , 310 , 330
  310 ZZ # Z(N)
      N # N - 1
      GO TO  340
  320 UPDOWN # 1.0
  330 ZZ # Z(N+1)
      N # N + 1
  340 CONTINUE
      IF ( ZZ - Z(NSTART) )  30  , 350 , 30
  350 KOUNT # KOUNT + 1
      IF ( KOUNT -  2 * NUM )  30 , 510 , 510
  360 A1 # SINF ( PHI(N) * CONV )
      B1 #  - AKK * COSF ( PHI(N) * CONV )
      IF ( ABSF ( XO - XD ) - .0001 )  370 , 370 , 380
  370 A2 # 1.0
      B2 # 0.0
      D2 # XO
      GO TO  390
  380 SLPE #  ( YD - YO ) / ( XD - XO )
      A2 # - SLPE
      B2 # 1.0
      D2 # YO - SLPE * XO
  390 DENOM # A1 * B2  -  A2 * B1
      IF ( ABSF ( DENOM ) - .0001 )   410 , 410 ,  400
  400 XI #  ( - B1 * D2 ) / DENOM
      YI #  A1 * D2 / DENOM
      XORE # XO + XC(N)
      XIRE # XI + XC(N)
      CALL  PLOT  ( XORE , YO , 1 , 2 )
      CALL  PLOT  ( XIRE , YI , 1 , 1 )
      XO # AKK * SQRTF ( XI**2 + YI**2 )
      YO # 0.0
      BETAZ # BETAZ - PHI(N)
      GO TO  220
  410 WRITE OUTPUT TAPE 6 , 4000 , N,AK(N),A1,B1,A2,B2,D2,PHI(N),XO,YO,
     1 XD, YD
 4000 FORMAT ( 60H1 LINE CONNECTING (XO,YO) AND (XD,YD) IS PARALLEL TO L
     1INE 2  /  1H0 , I3 ,  11F10.4  )
      GO TO  510
  420 IF ( UPDOWN )   430 , 430 , 440
  430 XO # R1(N)
      XD # XC(N)
      ZZ # Z(N)
      NN # N - 1
      GO TO  450
  440 XO # XC(N)
      XD # R1(N)
      ZZ # Z(N+1)
      NN # N + 1
  450 YO # THETA * R(N) * CONV
      YD # YO + R(N) * DTHETA(N) * CONV
      SLPE # ( YD - YO ) / ( XD - XO )
```

```
  460 CONTINUE
      IF ( YD - R2(N) )  480  , 480 , 470
  470 YI # R2(N)
      XI # ( SLPE * XO  + YI - YO )  / SLPE
      CALL  PLOT ( XO , YO , 1 , 2 )
      CALL  PLOT ( XI , YI , 1 , 1 )
      XO # XI
      YO # 0.0
      YD # YD - R2(N)
      GO TO  460
  480 CALL  PLOT ( XO , YO , 1 , 2 )
      CALL  PLOT ( XD , YD , 1 , 1 )
      THETA #  YD  /  ( R(N) * CONV )
      IF ( ZZ -  Z(NSTART)  )   500 , 490 , 500
  490 KOUNT # KOUNT + 1
      IF ( KOUNT - 2 * NUM )  500 , 510 , 510
  500 N # NN
      GO TO  30
  510 CONTINUE
      CALL  FINISH ( 30,TITLE )
      END FILE 8
      RETURN
      END
*LABEL


CDIVSUR                DIVIDE UP SURFACE
      SUBROUTINE  DIVSUR  ( STEP )
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      DIMENSION HANGLE(100),THNESS(100)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      THMAX # 0.0
      J # 0
      S # 0.0
      MM1 # M - 1
      DO  100  N # 1 , MM1
      IF ( AK(N) )   5 , 50 ,  5
    5 S # XI(N)
      AKOFN # AK(N) / F(N)
      RPX # R(N) - AKOFN * XI(N)
      ZPX # Z(N) - XI(N) / F(N)
   10 J # J + 1
      RR # RPX + AKOFN * S
      ZZ # ZPX + S / F(N)
   15 SUMTH # 0.0
      DO  20   I # 1 , NOGEOD
      CALL  PERCOV  ( RR , N , I , PERCNT , HANGL )
      HANGLE(I) # HANGL
      THNESS(I) # FLOATF ( NC(I) ) * D(I) * PERCNT
      SUMTH # SUMTH + THNESS(I)
   20 CONTINUE
```

```
      THMAX # MAXIF ( THMAX , SUMTH )
      RFINAL # RR + SUMTH / F(N)
      ZFINAL # ZZ - SUMTH * AKOFN
      WRITE TAPE 1 , S , RR , ZZ  , ( HANGLE(I) , I # 1, NOGEOD ) ,
     1 (THNESS(I) , I # 1 , NOGEOD ) , SUMTH , RFINAL , ZFINAL
      IF ( S - XI(N+1) + STEP ) 25 , 30 , 30
   25 S # S + STEP
      GO TO  10
   30 IF ( S - XI(N+1) + .000001 )  35 , 100 , 100
   35 S # XI(N+1)
      RR # R(N+1)
      ZZ # Z(N+1)
      J # J + 1
      GO TO  15
   50 J # J + 1
      S # XI(N)
      RR # R(N)
      ZZ # Z(N)
      SUMTH # 0.0
      DO  80  I # 1 , NOGEOD
      CALL  PERCOV ( RR , N , I , PERCNT , HANGL )
      HANGLE(I) # HANGL
      THNESS(I) # D(I) * PERCNT * FLOATF ( NC(I) )
      SUMTH # SUMTH + THNESS(I)
   80 CONTINUE
      THMAX # MAXIF ( THMAX , SUMTH )
      RFINAL # RR + SUMTH
      ZFINAL # ZZ
      J2 # 1
   90 WRITE TAPE 1 , S , RR , ZZ , ( HANGLE(I) , I # 1 , NOGEOD ) ,
     1 ( THNESS(I), I # 1 , NOGEOD ) , SUMTH , RFINAL , ZFINAL
      IF ( J2 - 2 )   95 , 100 , 100
   95 S # XI(N+1)
      J # J + 1
      ZZ # Z(N+1)
      ZFINAL # ZZ
      J2 # 2
      GO TO  90
  100 CONTINUE
      JJ # J
      SMAX # S
      END FILE 1
      RETURN
      END
*LABEL


CRZPLOT              PLOT R AND Z VERSUS S
      SUBROUTINE    RZPLOT   ( XO , YO , XL , YL )
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      DIMENSION  XAX(12)   ,   A(12)  ,   Y(210)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
```

```
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
B      TITLE(7) # 606060605161
B      TITLE(8) # 514421676060
B      TITLE(9) # 214524607161
B      TITLE(10)# 714421676060
B      TITLE(11)# 536060606060
B      XAX(1) #   606060606261
B      XAX(2) #   624421676060
B      XAX(3) #   536060606060
       DIVX # 10.0 / XL
       DIVY # 10.0 / YL
       CALL SETUP  ( TITLE )
       CALL AXPLOT ( XO,YO,XL , YL ,DIVX,DIVY, .1,.1, 5,5,50,XAX ,
     1 TITLE(7)  )
       INK # 2
       DO  3  N # 1 , M
       YI # R(N) * YL / RMAX + YO
       SI # XI(N) * XL / SMAX + XO
       CALL PLOT ( SI , YI , 1 , INK )
       INK # 1
     3 CONTINUE
       INK # 2
       DO  4  N # 1 , M
       YI # Z(N) * YL / ZMAX + YO
       SI # XI(N) * XL / SMAX + XO
       CALL  PLOT ( SI , YI , 1,INK )
       INK # 1
     4 CONTINUE
       WRITE OUTPUT TAPE 0,4010, XAX(2) , SMAX
  4010 FORMAT ( A6 , 1H# , F8.4 , 1H$ )
       READ INPUT TAPE 0,4020, (A(K), K # 1,3 )
  4020 FORMAT ( 3A6 )
       XX # XL / 4.0  + XO
       YY # YL + .5   + YO
       CALL LETTER ( XX , YY , 50 , 52 , A )
       WRITE OUTPUT TAPE 0,4010, TITLE(8), RMAX
       READ INPUT TAPE 0,4020, ( A(K) , K# 1, 3 )
       YY # YL + .3    + YO
       CALL LETTER ( XX, YY, 50 , 52 , A )
       WRITE OUTPUT TAPE 0,4010 , TITLE(10) , ZMAX
       READ INPUT TAPE 0,4020,( A(K) , K # 1 , 3 )
       YY # YL + .1    + YO
       CALL LETTER ( XX, YY, 50, 52 , A )
       YY # YO + YL + 2.0
       CALL LETTER ( XO, YY, 50, 52 , TITLE )
       CALL FINISH ( 30, TITLE )
       END FILE 8
       RETURN
       END
*LABEL

CANGLPL              PLOT HELIX ANGLE VERSUS S
       SUBROUTINE  ANGLPL ( XO , YO , XL , YL )
       DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
```

```
      1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
      2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
       DIMENSION  XAX(12) , A(12) ,  Y(210)
       COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
      1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
      2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
B      TITLE(7) #  603025433167
B      TITLE(8) #  602145274325
B      TITLE(9) #  536060606060
B      XAX(1)   #  606060606261
B      XAX(2)   #  624421676060
B      XAX(3)   #  536060606060
       DIVX # 10.0 / XL
       DIVY #  9.0 / YL
       CALL SETUP ( TITLE )
       CALL AXPLOT ( XO,YO,XL,YL,DIVX,DIVY,.1 ,10.0 , 5, 3 , 50 , XAX ,
      1 TITLE(7)  )
       DO  20   I # 1 , NOGEOD
       INK # 2
       REWIND 1
       DO  20   J # 1 , JJ
       I2 # I + 2
       READ TAPE 1 , S , ( Y(K) , K # 1 , I2 )
       IF ( Y(I2) )  20 , 20  , 5
     5 YI # Y(I2) * YL  / 90.0  + YO
       SI # S * XL / SMAX   + XO
       CALL  PLOT  ( SI , YI , 1 , INK )
       INK # 1
    20 CONTINUE
       WRITE OUTPUT TAPE 0 , 4010 , XAX(2) , SMAX
  4010 FORMAT ( A6 , 1H# , F8.4 , 1H$ )
       READ INPUT TAPE 0,4020, ( A(K), K # 1,3 )
  4020 FORMAT ( 3A6 )
       XX # XO + XL / 4.0
       YY # YO + YL + .1
       CALL LETTER ( XX , YY , 50 , 52 , A  )
       YY # YO + YL  + 2.0
       CALL LETTER ( XO , YY , 50, 52 , TITLE )
       CALL FINISH ( 30, TITLE )
       END FILE 8
       REWIND 1
       RETURN
       END
*LABEL

CTHPLOT        PLOT THICKNESS VS S FOR GEODESIC I
       SUBROUTINE   THPLOT (I , XO , YO , XL , YL )
       DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
      1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
      2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
       DIMENSION  XAX(12) , A(12) , B(12)   ,  Y(210)
       COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
      1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
      2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
```

```
B      XAX(1) #  606060626061
B      XAX(2) #  606244216760
B      XAX(3) #  536060606060
B      TITLE(7)  # 606063606160
B      TITLE(8)  # 634421676026
B      TITLE(9)  # 465160272546
B      TITLE(10) # 242562312360
       DIVX #   10.0 / XL
       DIVY #   10.0 / YL
       XLD #   .1
       YLD #   .1
       NX #   5
       NY #   5
       II #   50
       WRITE OUTPUT TAPE 0,1015 , XAX(2) , SMAX
  1015 FORMAT ( A6 , 1H# , F8.4 , 1H$ )
       READ INPUT TAPE 0,1020, ( A(K), K # 1,3 )
  1020 FORMAT ( 3A6 )
       WRITE OUTPUT TAPE 0,1025, THMAX
  1025 FORMAT ( 20HMAXIMUM THICKNESS # ,F9.6 , 1H$ )
       READ INPUT TAPE 0,1030, ( B(K) , K # 1, 5 )
  1030 FORMAT ( 5A6 )
    50 WRITE OUTPUT TAPE 0, 1005 , I
  1005 FORMAT ( I2 , 1H$ )
       READ INPUT TAPE 0,1010, TITLE(11)
  1010 FORMAT ( A6 )
    60 REWIND 1
       CALL  SETUP( TITLE )
       CALL  AXPLOT (XO,YO,XL,YL,DIVX,DIVY,XLD,YLD,NX,NY,II,XAX,TITLE(7))
       INK # 2
       DO  70  J # 1 , JJ
       I2 # I + NOGEOD + 2
       READ TAPE 1, S, ( Y(K) , K # 1 , I2 )
       SI # S * XL  / SMAX  + XO
       YI # Y(I2) * YL / THMAX  + YO
       CALL PLOT ( SI, YI , 1 , INK )
    70 INK # 1
       XX # XO  + XL / 4.0
       YY # YO  + YL  + .3
       CALL  LETTER ( XX, YY , 50 , 52 , A )
       YY # YO + YL + .1
       CALL LETTER ( XX, YY, 50 , 52 , B )
       YY #  YO + YL +  2.0
       CALL  LETTER  ( XO , YY , 50 , 52 , TITLE  )
       CALL FINISH ( 30 , TITLE )
    80 END FILE 8
       REWIND 1
       RETURN
       END
*LABEL


CSUMPLT      PLOT SUM OF THICKNESS VERSUS S
       SUBROUTINE   SUMPLT   ( XO , YO , XL , YL  )
       DIMENSION R(1000),Z(1000),AK(1000),F(1000),X1(1000),TITLE(12),
```

```
      1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
      2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
       DIMENSION  XAX(12) , A(12) , B(12)  ,  Y(110)
       COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
      1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
      2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
B      XAX(1) #  606060626061
B      XAX(2) #  606244216760
B      XAX(3) #  536060606060
B      TITLE(7)  # 606063606160
B      TITLE(8)  # 634421676026
B      TITLE(9)  # 465160272546
B      TITLE(10) # 242562312360
B   55 TITLE(11) #  606264445360
       DIVX #   10.0 / XL
       DIVY #   10.0 / YL
       XLD #   .1
       YLD #   .1
       NX #  5
       NY #  5
       II #  50
       WRITE OUTPUT TAPE 0,1015 , XAX(2) , SMAX
  1015 FORMAT ( A6 , 1H# , F8.4 , 1H$ )
       READ INPUT TAPE 0,1020, ( A(K), K # 1,3 )
  1020 FORMAT ( 3A6 )
       WRITE OUTPUT TAPE 0,1025, THMAX
  1025 FORMAT ( 20HMAXIMUM THICKNESS # ,F9.6 , 1H$ )
       READ INPUT TAPE 0,1030, ( B(K) , K # 1, 5 )
  1030 FORMAT ( 5A6 )
       REWIND 1
       CALL  SETUP( TITLE )
       CALL  AXPLOT (XO,YO,XL,YL,DIVX,DIVY,XLD,YLD,NX,NY,II,XAX,TITLE(7))
       INK # 2
       DO  70  J # 1 , JJ
       I2 # 2 * NOGEOD + 3
       READ TAPE 1, S, ( Y(K) , K # 1 , I2 )
       SI # S * XL  / SMAX  + XO
       YI # Y(I2) * YL / THMAX  + YO
       CALL PLOT ( SI, YI , 1 , INK )
    70 INK # 1
       XX # XO  + XL / 4.0
       YY # YO  + YL  + .3
       CALL  LETTER ( XX, YY , 50 , 52 , A )
       YY # YO + YL + .1
       CALL LETTER ( XX, YY, 50 , 52 , B )
       YY #  YO + YL +  2.0
       CALL  LETTER  ( XO , YY , 50 , 52 , TITLE  )
       CALL FINISH ( 30 , TITLE )
       END FILE 8
       REWIND 1
       RETURN
       END
*LABEL
```

```
CCNPLOT                   PLOT OF FINAL CONTOUR      R VS Z
      SUBROUTINE   CNPLOT   ( XO , YO , XSCALE , YSCALE )
C
C     XSCALE  IS SCALE OF X ( Z ) AXIS          ( 1.0 FULL SCALE , .5 HALF
C     YSCALE  IS SCALE OF Y ( R ) AXIS           SCALE, 2.0 DOUBLE, ETC. )
C
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      DIMENSION  A(12) ,  Y(210)   , XAX(12) , YAX(12)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
B     TITLE(7) #  602631452143
B     TITLE(8) #  602346456346
B     TITLE(9) #  645153606060
B     XAX(1) # 606071536060
B     YAX(1) # 606051536060
      CALL  SETUP ( TITLE )
      XL # ( ZMAX + THMAX + .1 ) * XSCALE
      YL # ( RMAX + THMAX + .1 ) * YSCALE
      DIVX  # 1.0 / XSCALE
      DIVY  # 1.0 / YSCALE
      NX # DIVX + .99
      NY # DIVY + .99
      CALL  AXPLOT ( XO,YO,XL,YL,DIVX,DIVY,1.0,1.0, NX,NY,50,XAX,YAX )
      INK # 2
      DO  280  N # 1 , M
      XX # Z(N) * XSCALE + XO
      YY # R(N) * YSCALE + YO
      CALL  PLOT ( XX , YY , 1 , INK )
      INK # 1
  280 CONTINUE
      REWIND 1
      IN1 # 2 * NOGEOD + 4
      IN2 # IN1 + 1
      INK # 2
      DO   300   J # 1 , JJ
      READ TAPE 1 , S , ( Y(K) , K # 1 , IN2 )
      XX # Y(IN2) * XSCALE + XO
      YY # Y(IN1) * YSCALE + YO
      CALL  PLOT ( XX , YY , 1 , INK )
      INK # 1
  300 CONTINUE
      XX # XO + XL / 4.0
      YY # YO + YL + .3
      WRITE OUTPUT TAPE 0,1000, XSCALE , YSCALE
 1000 FORMAT ( 9HZ SCALE # , F7.4 , 12H   R SCALE # , F7.4 , 1H$ )
      READ INPUT TAPE 0,1001, ( A(I) , I # 1 ,6 )
 1001 FORMAT ( 6A6 )
      CALL  LETTER  ( XX , YY , 50, 52 , A )
      YY # YO + YL + 2.0
      CALL  LETTER  ( XO , YY , 50 , 52 , TITLE  )
      CALL  FINISH ( 30 , TITLE )
      END FILE 8
```

```
      REWIND 1
      RETURN
      END
*LABEL

CPERCOV    COMPUTE PERCENT COVERAGE AND HELIX ANGLE AT A STATION
      SUBROUTINE  PERCOV  ( RR , N , I , PERCNT , HANGL )
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      IF ( AK(N) )   60 ,  162 , 60
   60 FOK # F(N) / ABSF (AK(N) )
      WO2 # W(I) / 2.0
      RHO # FOK * RR
      IF ( RR - CONS(I) )  140 , 130 , 100
  100 RMC # SQRTF ( ( RR-CONS(I))* (RR+CONS(I) ) )
      HANGL # ATANF ( CONS(I) / RMC ) * 180.0 / PI
      SINA # CONS(I) / RR
      COSA # RMC / RR
      F1 # RHO * SINA + WO2
      FAC # ( RHO + F1 ) * ( RHO - F1 )
      IF ( FAC )  120 , 120 , 110
  110 Y2 # - COSA * F1 + SINA * SQRTF ( FAC )
      X2 # COSA * Y2 / SINA  +  WO2 / SINA
      DPHI2 # ATANF ( ABSF( Y2 ) / ( RHO + X2 )   )
      F2 # RHO * SINA  -  WO2
      FAC2 # ( RHO + F2 ) * ( RHO - F2 )
      FAC2 # MAXIF ( FAC2 , 0.0 )
      Y3 # -COSA * F2  + SINA * SQRTF ( FAC2 )
      X3 # COSA * Y3 / SINA  -  WO2 / SINA
      DPHI1 #  ATANF ( Y3 / ( RHO + X3 )  )
      PERCNT # FOK * ( DPHI1 + DPHI2 ) / PI
      GO TO  170
  120 F3 # RHO * SINA - WO2
      FAC3 # ( RHO + F3 ) * ( RHO - F3 )
      FAC3 # MAXIF ( FAC3 , 0.0 )
      Y3 # - COSA * F3 + SINA * SQRTF ( FAC3 )
      X3 # COSA * Y3  / SINA  - WO2 / SINA
      DPHI1 # ATANF (  Y3 / ( RHO + X3 ) )
      Y2 # - COSA * F3 - SINA * SQRTF ( FAC3 )
      X2 # COSA * Y2 / SINA  - WO2 / SINA
      RHOX2 # RHO + X2
      IF ( RHOX2 )  122 , 122 , 126
  122 DPHI2 # ATANF ( ABSF( RHOX2 / Y2 )  )
      DPHI2 # DPHI2 + PI / 2.0
      GO TO 128
  126 DPHI2 # ATANF ( ABSF(Y2) / RHOX2 )
  128 PERCNT # FOK * ( DPHI1 + DPHI2 ) / ( 2.0 * PI )
      GO TO  170
  130 HANGL #  90.0
      GO TO  160
```

```
  140 HANGL # 0.0
      RHOMIN # FOK * CONS(I) - WO2
      IF ( RHO - RHOMIN )  150 ,  160 , 160
  150 PERCNT # 0.0
      GO TO  170
  160 RHOT # FOK * CONS(I)
      F4 # RHOT - WO2
      FAC4 # ( RHO - F4 ) * ( RHO + F4 )
      IF ( FAC4 )  150 , 150 , 161
  161 YI # SQRTF ( FAC4 )
      DPHII # ATANF ( YI / F4 )
      PERCNT # FOK * DPHII / PI
      GO TO  170
  162 IF ( RR - CONS(I) )  168 , 168 , 164
  164 RMC # SQRTF (  (RR+CONS(I) ) * ( RR-CONS(I) )  )
      HANGL # ATANF ( CONS(I) / RMC ) * 180.0 / PI
      PERCNT # W(I) / ( PI * RMC )
      GO TO  170
  168 HANGL # 0.0
      PERCNT # 0.0
  170 CONTINUE
      RETURN
      END
*LABEL


CADJUST   ADJUST STARTING HELIX ANGLE
      SUBROUTINE  ADJUST ( I, NA , NB , FRACT , EPS , LL )
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),XI(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      COMMON M,R,Z,AK,F,XI,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      CONV # 180.0 / PI
      ITER # 0
      AZERO # ALPHAO(I)
      RZERO # RO(I)
      ZZERO # ZO(I)
      C # CONS(I)
      IF ( ALPHAO(I) - 89.0 )  30 , 30 , 20
   20 RO(I) # RMAX
      ALPHAO(I) # ATANF( CONS(I) / SQRTF( RO(I)**2 - CONS(I)**2) ) *CONV
      DO 22  N # 2 , M
      IF ( RO(I) - R(N) )  24 , 24 , 22
   22 CONTINUE
   24 ZO(I) # Z(N)
   30 FRC# FRACT
      RV # FLOATF( NA ) / FLOATF ( NB) + ADVNCE
      AAZERO # ALPHAO(I)
   40 CONTINUE
      DELA# RV - FRC
      IF ( ABSF( DELA) - EPS ) 110 ,110 , 50
   50 DTDA # 0.0
      CSQ # CONS(I)**2
```

```
      RCOS # RO(I) * COSF ( ALPHAO(I) / CONV )
      NLI # NLOW + 1
      NHI # NHIGH - 1
      SQ2 #  1.0 / SQRTF ( R(NLI)**2 - CSQ )
      DTDA # DTDA - F(NLOW) * RCOS * SQ2 / AK(NLOW)
      IF ( NHI - NLI )  85 , 55 , 55
   55 DO  80   N # NLI , NHI
      IF ( AK(N) )  60 , 70 , 60
   60 SQ1 # SQ2
      SQ2 # 1.0 / SQRTF ( R(N+1)**2 - CSQ )
      DTDA # DTDA + F(N) * RCOS * ( - SQ2 + SQ1 ) / AK(N)
      GO TO  80
   70 DTDA # DTDA + RCOS * R(N) * ( Z(N+1) - Z(N) ) * ( SQ2 **3 )
   80 CONTINUE
   85 DTDA # DTDA + F(NHIGH) * RCOS * SQ2 / AK(NHIGH)
      DTDA # 2.0 * DTDA
      IF ( ABSF( DTDA ) - .01 )  140 , 140 , 90
   90 DALPHA # DELA* 360.0 / DTDA
      ALPHAO(I) # ALPHAO(I) + DALPHA
      IF ( ITER - 10 )  100 , 150 , 150
  100 ITER # ITER + 1
      CONS(I) # RO(I) * SINF ( ALPHAO(I) / CONV )
      CALL  DELTHA ( I )
      NLOW # NLOW
      NHIGH # NHIGH
      FRC # TSUM / 360.0
      GO TO  40
  110 DALPHA # ALPHAO(I) - AAZERO
      IF ( ABSF ( DALPHA ) - 5.0 )  120 , 130 , 130
  120 LL # 0
      GO TO  170
  130 WRITE OUTPUT TAPE 6 , 1000 , DALPHA
 1000 FORMAT ( 20H0 CHANGE IN ALPHA ,     , F10.6 , 43H , TOO GREAT - GEOD
     1ESIC DISTORTED INSTEAD     )
      GO TO  160
  140 WRITE OUTPUT TAPE 6 , 1010 , DTDA
 1010 FORMAT ( 22H0 D THETA / D ALPHA #   ,  F9.6 , 71H , LARGE CHANGE IN
     1ALPHA WOULD BE REQUIRED - GEODESIC DISTORTED INSTEAD  )
      GO TO  160
  150 WRITE OUTPUT TAPE 6, 1020
 1020 FORMAT (  71H0 ALPHA DID NOT CONVERGE IN 10 ITERATIONS - GEODESIC
     1DISTORTED INSTEAD )
  160 LL # 1
      CONS(I) # C
      RO(I) # RZERO
      ZO(I) # ZZERO
      ALPHAO(I) # AZERO
      CALL  DELTHA ( I )
  170 CONTINUE
      RETURN
      END
*LABEL
```

CNOFACT   ALTERS FRACTION SO NO COMMON FACTORS

```
      SUBROUTINE  NOFACT ( NUMER , IDENOM )
  100 JJJ # 0
  200 JJJ # JJJ + 1
      M1 # IGCD ( NUMER,IDENOM )
      IF ( M1 - 1 )    300 , 300 ,  250
  250 GO TO ( 1 , 2, 3, 4, 5 ) , JJJ
    1 IDENOM # IDENOM + 1
      GO TO 200
    2 IDENOM # IDENOM - 2
      GO TO 200
    3 IDENOM # IDENOM + 1
      NUMER # NUMER + 1
      GO TO 200
    4 NUMER # NUMER - 2
      GO TO 200
    5 NUMER # NUMER + 2
      IDENOM # IDENOM + 1
      GO TO 100
  300 CONTINUE
      RETURN
      END

      FUNCTION IGCD(MM,NN)
C PROGRAM AUTHOR M.ELSON,                   CENTRAL DATA PROCESSING,1/1/65
      M#MM
      N#NN
      IF(M-N)2,2,1
    1 I#M
      M#N
      N#I
    2 IGCD#M
      IGCD1#XMODF(N,M)
      IF(IGCD1)4,4,3
    3 N#M
      M#IGCD1
      GOTO2
    4 RETURN
      END
*LABEL
CSHIFT
      SUBROUTINE  SHIFT  ( N , SHIFT2 )
      DIMENSION R(1000),Z(1000),AK(1000),F(1000),X1(1000),TITLE(12),
     1RO(100),ZO(100),ALPHAO(100),CONS(100),W(100),D(100),THICK(100),
     2RT(100),ZT(100),NC(100),DTHETA(1000),FLNGTH(1000)
      DIMENSION R1(100 ) , R2(100 ) , PHI(100 ) , XC(100 )
      COMMON M,R,Z,AK,F,X1,NOGEOD,RO,ZO,ALPHAO,CONS,W,D,THICK,NC,RT,ZT,
     1SMAX,RMAX,ZMAX,THMAX,JJ,TITLE,PI,DTHETA,FLNGTH,TSUM,FLSUM,NHIGH,
     2 NLOW , DISTRT , ADVNCE , SHAFT1 , SHAFT2
      COMMON  AA,BB,CC,DEL,DELRHO,NSTART
      COMMON  LLL , RHOMIN , FR , TMIN
      COMMON  R1 , R2 , PHI , XC
      SHIFT2 # 0.0
      I # N - 1
    5 IF ( AK(I) )    10 , 60 , 10
```

```
   10 IF ( PHI(I) - 90.0 )  50  , 50 , 20
   20 IF ( PHI(I) - 180.0 )  30 , 40 , 40
   30 SHIFT2 # SHIFT2 + R1(I) - R2(I) * COSF ( PHI(I) * PI / 180.0 )
      GO TO 60
   40 SHIFT2 # SHIFT2 + R1(I) + R2(I)
      GO TO 60
   50 SHIFT2 # SHIFT2 + R1(I) * ( 1.0 - COSF ( PHI(I) * PI / 180.0 ) )
   60 IF ( I - N )  70 , 80 , 80
   70 I # N
      GO TO  5
   80 CONTINUE
      RETURN
      END
*LABEL

CAXPLOT                          DRAW AXES FOR PLOTS
      SUBROUTINE  AXPLOT  ( XO,YO,XL,YL,DIVX,DIVY,XLD,YLD,NX,NY,II,XAX,
     1 YAX )
C
C     XO , YO  IS THE ORIGIN
C     XL , YL  IS LENGTH OF AXES
C     DIVX,DIVY IS DIVISIONS PER INCH
C     XLD, YLD IS LENGTH DIVISION REPRESENTS
C     NX , NY  IS DIVISIONS TO BE LABELED ( 1,EVERY DIV , 2,EVERY OTHER)
C     II  IS SIZE OF LETTERS
C     XAX , YAX IS NAME OF AXES
C
      DIMENSION  XAX(12) , YAX(12)
      XXL # XL + XO
      YYL # YO + YL
      CALL PLOT ( XO ,YYL , 1 , 2)
      CALL PLOT ( XO , YO , 1 , 1)
      IX # DIVX * XL  + .05
      IY # DIVY * YL  + .05
      YOFF1 # YO + .04
      YOFF2 # YO - .04
      XOFF1 # XO - .04
      XOFF2 # XO + .04
      DO 20 I # 1 , IY
      YI # I
      YYI # YI / DIVY + YO
      CALL PLOT ( XOFF1 , YYI , 1 , 2 )
   20 CALL PLOT ( XOFF2 , YYI , 1 , 1 )
      CALL PLOT ( XXL, YO , 1 , 2)
      CALL PLOT ( XO , YO , 1 , 1)
      DO 10  I # 1 , IX
      XI # I
      XXI # XI / DIVX + XO
      CALL PLOT ( XXI , YOFF1 , 1 , 2 )
   10 CALL PLOT ( XXI , YOFF2 , 1 , 1 )
      IF ( II - 51 )  30 , 40 , 50
   30 SIZE # .096
      GO TO 90
   40 SIZE # .192
```

```
      GO TO 90
   50 IF ( II - 53 )  60 , 70 , 80
   60 SIZE # .384
      GO TO 90
   70 SIZE # .768
      GO TO 90
   80 SIZE # 1.536
   90 CONTINUE
      YOFF # YOFF2 - SIZE - .1
      DO  100  I # NX , IX , NX
      XI # I
      XXI # XI / DIVX
      XXXI # XI * XLD
      WRITE OUTPUT TAPE  0,1000,XXXI
 1000 FORMAT ( F5.2 , 1H$ )
      READ INPUT TAPE 0,1002, A
 1002 FORMAT ( A6 )
      XXI # XXI - 2.5 * SIZE  + XO
  100 CALL LETTER ( XXI , YOFF , II , 52 , A )
      YOFF # YOFF - SIZE - .1
      XX # XO + XL / 4.0
      CALL LETTER ( XX , YOFF , II , 52 , XAX )
      XOFF # XOFF1 - .05 - 5.0 * SIZE
      DO  110  I #  NY , IY , NY
      YI # I
      YYI # YI / DIVY + YO - SIZE / 2.0
      YYYI # YI * YLD
      WRITE OUTPUT TAPE 0,1000, YYYI
      READ INPUT TAPE 0, 1002 , A
  110 CALL LETTER ( XOFF , YYI , II , 52 , A )
      XOFF # XOFF - SIZE - .1
      YY # YO + YL / 4.0
      CALL LETTER ( XOFF , YY , II , 53 , YAX )
      RETURN
      END
```

```
MAC1   #MACRO/M                                                      1001
$$      M IS THE NUMBER OF POINTS DEFINING THE CONTOUR              1002
        N#0                                                          1003
        X1(1)#0                                                      1005
1010)   N#N+1                                                        1007
        IF(ABSF(Z(N)-Z(N+1))-.000001)1011,1011,1018                  1028
1011)   IF(R(N+1)-R(N))1012,1012,1013                                1030
```

```
1012)  K(N)#-(10**20)
       R1(N)#R(N+1)
       R2(N)#R(N)
       XC(N)#X1(N)+R2(N)
       JUMPTO/1014
1013)  K(N)#10**20
       R1(N)#R(N)
       R2(N)#R(N+1)
       XC(N)#X1(N)-R1(N)
1014)  F(N)#10**20
       PHI(N)#359.9
       X1(N+1)#X1(N)+R2(N)-R1(N)
       X2(N)#X1(N+1)
       JUMPTO/1050
1018)  K(N)#(R(N+1)-R(N))/(Z(N+1)-Z(N))
       F(N)#SQRTF(1+K(N)**2)
       X1(N+1)#X1(N)+(Z(N+1)-Z(N))*F(N)
       X2(N)#X1(N+1)
       PHI(N)#ABSF(K(N)/F(N))*360
       IF(ABSF(K(N))-.000001)1040,1040,1020
1020)  IF(K(N))1025,1025,1030
1025)  XC(N)#X1(N+1)-R(N+1)*F(N)/K(N)
       R1(N)#XC(N)-X1(N+1)
       R2(N)#XC(N)-X1(N)
       JUMPTO/1050
1030)  XC(N)#X1(N)-R(N)*F(N)/K(N)
       R1(N)#X1(N)-XC(N)
       R2(N)#X1(N+1)-XC(N)
       JUMPTO/1050
1040)  R2(N)#6.283181*R(N)
       K(N)#0
       XC(N) # 0.0
1050)  IF(N-M+1.1)1010,1060,1060
1060)  N#1
1070)  N#N+1
       IF(K(N)-K(N-1))1080,1080,1090
1080)  IF(N-M+1.1)1070,1200,1200
1090)  SHIFT#0
       I#N-1
1100)  IF(K(I))1110,1160,1110
1110)  IF(PHI(I)-90)1120,1120,1130
1120)  SHIFT#SHIFT+R1(I)*(1-COSF(PHI(I)))
       JUMPTO/1160
1130)  IF(PHI(I)-180)1140,1150,1150
1140)  SHIFT#SHIFT+R1(I)-R2(I)*COSF(PHI(I))
       JUMPTO/1160
1150)  SHIFT#SHIFT+R1(I)+R2(I)
1160)  IF(I-N+.1)1170,1180,1180
1170)  I#N
       JUMPTO/1100
1180)  I#N-1
1190)  I#I+1
       X1(I)#X1(I)+SHIFT
       X2(I)#X2(I)+SHIFT
       XC(I)#XC(I)+SHIFT
```

```
       IF(I-M+1.1)1190,1070,1070                                1313
1200)  L1#LINE/0,0,10,0                                         1401
       N#0                                                      1403
1210)  N#N+1                                                    1405
       IF(K(N))1220,1240,1230                                   1407
1220)  PC(N)#POINT/XC(N),0                                      1409
       L2(N)#LINE/PC(N),ATANGL,(180-PHI(N))                     1411
       C1(N)#CIRCLE/CENTER,PC(N),RADIUS,R1(N)                   1413
       C2(N)#CIRCLE/CENTER,PC(N),RADIUS,R2(N)                   1415
       JUMPTO/1250                                              1417
1230)  PC(N)#POINT/XC(N),0                                      1419
       L2(N)#LINE/PC(N),ATANGL,PHI(N)                           1421
       C1(N)#CIRCLE/CENTER,PC(N),RADIUS,R1(N)                   1423
       C2(N)#CIRCLE/CENTER,PC(N),RADIUS,R2(N)                   1425
       JUMPTO/1250                                              1501
1240)  L2(N)#LINE/PARLEL,L1,YLARGE,R2(N)                        1503
       L3(N)#LINE/(POINT/X1(N),0),PERPTO,L1                     1505
       L4(N)#LINE/(POINT/X2(N),0),PERPTO,L1                     1507
1250)  IF(N-M+1.1)1210,1260,1260                                1509
1260)  TERMAC                                                   1511

MAC2  #MACRO/M                                                  2001
$$     M IS THE NUMBER OF POINTS DEFINING THE CONTOUR           2002
       O#POINT/0,0                                              2005
       STRT#POINT/0,10                                          2007
       TLON                                                     2011
       FROM/STRT                                                2013
       GOTO/O                                                   2015
       DRAFT/ON                                                 2009
       N#0                                                      2017
2010)  N#N+1                                                    2019
       IF(PHI(N)-180) 2012,2012,2014                            2025
2012)  II#1                                                     2027
       JUMPTO/2016                                              2029
2014)  II#2                                                     2031
2016)  IF(K(N))2020,2025,2030                                   2021
2020)  GOBACK/C2(N) ,ON,II,INTOF,L2(N)                          2023
2022)  GORGT/L2(N) ,ON,C1(N)                                    2101
       TLON,GORGT/C1(N),TO,II,INTOF,L1                          2103
       JUMPTO/2040                                              2105
2025)  DNTCUT                                                  A2130
       GODLTA/.1,0,0                                           B2130
       INDIRV/-1,0,0                                           C2130
       GO/ON,L3(N)                                             D2130
       CUT                                                     E2130
       GORGT/L3(N),ON,L2(N)                                    F2130
2027)  GORGT/L2(N) ,ON,L4(N)                                    2132
       TLON,GORGT/L4(N),TO,L1                                   2134
       JUMPTO/2040                                              2136
2030)  GOBACK/C1(N) ,ON,II,INTOF,L2(N)                          2107
2035)  GORGT/L2(N) ,ON,C2(N)                                    2109
       TLON,GORGT/C2(N),TO,II,INTOF,L1                          2111
2040)  IF(N-M+1.1)2050,2070,2070                                2113
2050)  IF(ABSF(X1(N+1)-X2(N))-.000001)2010,2010,2055            2115
```

```
2055)  GOTO/(POINT/XI(N+1),0)                                                  2117
       N#N+1                                                                   2119
       IF(PHI(N)-180) 2057,2057,2059                                           2160
2057)  II#1                                                                    2162
       JUMPTO/2060                                                             2164
2059)  II#2                                                                    2166
2060)  IF(K(N))2062,2064,2066                                                  2140
2062)  GOLFT/C2(N) ,ON,II,INTOF,L2(N)                                          2142
       JUMPTO/2022                                                             2144
2064)  GOLFT/L3(N) ,ON,L2(N)                                                   2146
       JUMPTO/2027                                                             2148
2066)  GOLFT/C1(N)  ,ON,II,INTOF,L2(N)                                         2150
       JUMPTO/2035                                                             2152
2070)  GOTO/0                                                                  2125
       DRAFT/OFF                                                               2201
       TERMAC                                                                  2203


MAC3  #MACRO/RO,AZERO,PRIME,M,EPS                                              3001
$$     RO IS THE RADIUS OF THE STARTING STATION                               A3000
$$     AZERO IS THE HELIX ANGLE AT THE STARTING STATION                       B3000
$$     PRIME IS THE DESIRED NUMBER OF CIRCUITS PER PATTERN ( A PRIME NO. )    C3000
$$     M IS THE NUMBER OF POINTS DEFINING THE CONTOUR                         D3000
$$     EPS IS THE MAXIMUM ALLOWABLE DIFFERENCE EETWEEN REVOLUTIONS            E3000
$$     PER CIRCUIT OBTAINED AND REVOLUTIONS PER CIRCUIT DESIRED               F3000
       N#0                                                                     3002
3010)  N#N+1                                                                   3003
       IF ( R(N)-RO )  3012,3018,3015                                          3004
3012)  IF ( N-M )  3010,3355,3355                                              3005
3015)  N#N-1                                                                   3006
3018)  J#N                                                                     3007
3020)  PASS#1                                                                  3011
       DEG#180/3.1415927                                                       3012
       ALPHA#AZERO                                                             3010
3030)  SINA#SINF(ALPHA)                                                        3013
       COSA#COSF(ALPHA)                                                        3015
       CONS#RO*SINA                                                            3017
       IF(CONS-R(1))3280,3040,3040                                             3019
3040)  I#J                                                                     3021
3050)  I#I-1                                                                   3023
       IF(R(I)-CONS)3060,3060,3050                                             3025
3060)  IF(CONS-R(M))3280,3070,3070                                             3103
3070)  L#J                                                                     3105
3080)  L#L+1                                                                   3107
       IF(R(L)-CONS)3090,3090,3080                                             3109
3090)  L#L-1                                                                   3111
       N#I                                                                     3113
       ASEC2#ATANF(SQRTF((R(N+1)/CONS)**2-1))                                  3115
       DBETA(N)#ASEC2                                                          3117
       DTHETA(N)#F(N)*DBETA(N)/K(N)                                            3119
       FLNGTH(N)#R2(N)*SINF(DBETA(N))                                          3120
3100)  N#N+1                                                                   3121
       IF(K(N))3120,3110,3120                                                  3123
3110)  DTHETA(N)#CONS*(Z(N+1)-Z(N))*DEG/(R(N)*SQRTF(R(N)**2-CONS**2))          3201
       FLNGTH(N)#SQRTF((Z(N+1)-Z(N))**2+(R(N)*DTHETA(N)/DEG)**2)               3202
```

```
        JUMPTO/3100                                                     3203
3120)   IF(N-L)3130,3140,3140                                           3205
3130)   ASEC1#ASEC2                                                     3207
        ASEC2#ATANF(SQRTF((R(N+1)/CONS)**2-1))                          3209
        DBETA(N)#ABSF(ASEC2-ASEC1)                                      3211
        DTHETA(N)#F(N)*DBETA(N)/ABSF(K(N))                              3213
        FLNGTH(N)#SQRTF(R1(N)**2+R2(N)**2-2*R1(N)*R2(N)*COSF(DBETA(N))) 3214
        JUMPTO/3100                                                     3215
3140)   DBETA(N)#ASEC2                                                  3217
        DTHETA(N)#F(N)*(-ASEC2)/K(N)                                    3219
        FLNGTH(N)#R2(N)*SINF(DBETA(N))                                  3220
        TSUM#0                                                          3221
        N#I-1                                                           3223
3150)   N#N+1                                                           3225
        TSUM#TSUM+DTHETA(N)                                             3301
        IF(N-L)3150,3160,3160                                           3303
3160)   TSUM#2*TSUM                                                     3305
        RVN#TSUM/360                                                    3307
        N#0                                                             3309
3170)   N#N+1                                                           3311
        IF(RVN-N)3180,3190,3170                                         3313
3180)   INTGER#N-1                                                      3315
        FRACT#RVN-INTGER                                                3317
        JUMPTO/3200                                                     3319
3190)   INTGER#N                                                        3321
        FRACT#0                                                         3323
3200)   N#1                                                             3325
        PARTN#1/PRIME                                                   3327
3210)   N#N+1                                                           3401
        PARTN1#PARTN                                                    3329
        PARTN#N/PRIME                                                   3331
        IF(FRACT-PARTN  )3230,3260,3220                                 3403
3220)   IF(N-PRIME+1)3210,3250,3250                                     3405
3230)   IF(ABSF(FRACT-PARTN  )-ABSF(FRACT-PARTN1    ))3250,3250,3240    3407
3240)   N#N-1                                                           3409
        PARTN#PARTN1                                                    3410
3250)   DEL#PARTN  -FRACT                                               3411
        IF(ABSF(DEL)-EPS)3270,3270,3290                                 3413
3260)   DEL#0                                                           3415
3270)   PRINT/3,PASS,DEL,N,PARTN  ,FRACT,INTGER,$                       3417
        RVN,ALPHA,TSUM,   CONS,              L,I                        3418
        N#I-1                                                           3430
3272)   N#N+1                                                           3432
        PRINT/3,DBETA(N),DTHETA(N) ,FLNGTH(N)                           3434
        IF(N-L)3272,3380,3380                                           3436
3280)   PRINT/0                                                         3423
TITLES  MINIMUM RADIUS IS LESS THAN R(1) OR R(M)                        3425
        JUMPTO/3380                                                     3501
3290)   IF(PASS-10)3295,3295,3360                                       3503
3295)   CSQ#CONS**2                                                     3505
        RCOS#RO*COSA                                                    3507
        N#I                                                             3508
        SQ2#1/SQRTF(R(I+1)**2-CSQ)                                      3509
        SUM#0                                                           3511
        SUM#SUM-F(I)*RCOS*SQ2/K(I)                                      3513
```

```
3300)  N#N+1                                                    3515
       IF(N-L+1)3310,3310,3340                                  3517
3310)  IF(K(N))3320,3330,3320                                   3519
3320)  SQ1#SQ2                                                  3521
       SQ2#1/SQRTF(R(N+1)**2-CSQ)                               3523
       SUM#SUM+F(N)*RCOS*(-SQ2+SQ1)/K(N)                        3525
       JUMPTO/3300                                              3601
3330)  SUM#SUM+RCOS*R(N)*(Z(N+1)-Z(N))*(SQ2**3)                 3603
       JUMPTO/3300                                              3605
3340)  SUM#SUM+F(L)*RCOS*SQ2/K(L)                               3607
       SUM#2*SUM                                                3609
       IF(ABSF(SUM)-.000001)3370,3370,3350                      3611
3350)  DALPHA#DEL/SUM *360                                      3613
       ALPHA#ALPHA+DALPHA                                       3615
       PASS#PASS+1                                              3617
       JUMPTO/3030                                              3619
3355)  PRINT/0                                                  3630
TITLES  RO AS GIVEN IS GREATER THAN R MAX OF PART               3634
       JUMPTO/3380                                              3636
3360)  PRINT/0                                                  3621
TITLES ALPHA DID NOT CONVERGE IN TEN PASSES                     3623
       JUMPTO/3380                                              3625
3370)  PRINT/0                                                  3701
TITLES CHANGE IN ALPHA WILL NOT CHANGE THETA                    3703
3380)  TERMAC                                                   3705

MAC4  #MACRO/TZERO,J,NUMBER                                     4001
$$     TZERO IS THE STARTING VALUE OF THETA                    A4002
$$     J IS THE STARTING SECTION FOR THE PLOT                  B4002
$$     NUMBER IS THE NUMBER OF CIRCUITS TO BE DRAWN            C4002
       N#J                                                      4003
       THETA#TZERO                                              4008
       ZZ#Z(J)                                                  4002
       RADIAN#3.1415927/180                                     4004
       ZFLAG#0                                                  4005
       TLON                                                     4006
4010)  IF(K(N))4020,4330,4050                                   4007
4020)  KK#-1                                                    4009
       IF(ABSF(ZZ-Z(N))-.000001)4030,4030,4040                  4011
4030)  RFLAG#2                                                  4013
       RR#R2(N)                                                 4015
       JUMPTO/4080                                              4017
4040)  RFLAG#1                                                  4019
       RR#R1(N)                                                 4021
       JUMPTO/4080                                              4023
4050)  KK#1                                                     4025
       IF(ABSF(ZZ-Z(N+1))-.000001) 4070,4070,4060               4101
4060)  RFLAG#1                                                  4103
       RR#R1(N)                                                 4105
       JUMPTO/4080                                              4107
4070)  RFLAG#2                                                  4109
       RR#R2(N)                                                 4111
4080)  BETA#KK*K(N)*THETA/F(N)                                  4113
       XO#KK*RR*COSF(BETA)                                      4115
```

```
       YO#RR*SINF(BETA)                                4117
       IF(KK)4090,4090,4110                            4119
4090)  IF(N-L+.1)4120,4100,4100                        4121
4100)  RE#RR                                           4123
       BETAZ#BETA+2*DBETA(N)                           4125
       JUMPTO/4160                                     4201
4110)  IF(N-I-.1)4100,4100,4120                        4203
4120)  IF(RFLAG-1.5)4130,4130,4140                     4205
4130)  RE#R2(N)                                        4207
       JUMPTO/4150                                     4209
4140)  RE#R1(N)                                        4211
4150)  BETAZ#BETA+DBETA(N)                             4213
4160)  XD#KK*RE*COSF(BETAZ)                            4215
       YD#RE*SINF(BETAZ)                               4217
       IF(BETAZ-PHI(N))4220,4170,4170                  4219
4170)  A1#SINF(PHI(N))                                 4221
       B1#-KK*COSF(PHI(N))                             4223
       IF(ABSF(XO-XD)-.000001)4180,4180,4190           4225
4180)  A2#1                                            4301
       B2#0                                            4303
       D2#XO                                           4305
       JUMPTO/4200                                     4307
4190)  SLPE #(YD-YO)/(XD-XO)                           4309
       A2#-SLPE                                        4311
       B2#1                                            4313
       D2#YO-SLPE *XO                                  4315
4200)  DENOM#A1*B2-A2*B1                               4317
       IF(ABSF(DENOM)-.000001)4420,4420,4210           4319
4210)  XI#-B1*D2/DENOM                                 4321
       YI#A1*D2/DENOM                                  4323
       XOREF#XO+XC(N)                                  4401
       XIREF#XI+XC(N)                                  4403
       GOTO/(POINT/XOREF,YO)                           4405
       DRAFT/ON                                        4407
       GOTO/(POINT/XIREF,YI)                           4409
       DRAFT/OFF                                       4411
       XO#KK*SQRTF(XI**2+YI**2)                        4413
       YO#0                                            4414
       BETAZ#BETAZ-PHI(N)                              4415
       JUMPTO/4160                                     4417
4220)  XOREF#XO+XC(N)                                  4419
       XDREF#XD+XC(N)                                  4421
       GOTO/(POINT/XOREF,YO)                           4423
       DRAFT/ON                                        4425
       GOTO/(POINT/XDREF,YD)                           4501
       DRAFT/OFF                                       4502
       THETA#KK*F(N)*BETAZ/K(N)                        4503
       IF(KK)4230,4230,4270                            4505
4230)  IF(N-L+.1)4240,4260,4260                        4507
4240)  IF(RFLAG-1.5)4260,4260,4250                     4509
4250)  ZZ#Z(N+1)                                       4511
       N#N+1                                           4513
       JUMPTO/4310                                     4515
4260)  ZZ#Z(N)                                         4517
       N#N-1                                           4519
```

```
        JUMPTO/4310
4270)   IF(N-I-.1)4300,4300,4280
4280)   IF(RFLAG-1.5)4300,4300,4290
4290)   ZZ#Z(N)
        N#N-1
        JUMPTO/4310
4300)   ZZ#Z(N+1)
        N#N+1
4310)   IF(ABSF(ZZ-Z(J))-.000001)4320,4320,4010
4320)   ZFLAG#ZFLAG+1
        IF(ZFLAG-2*NUMBER+.1)4010,4010,4430
4330)   IF(ABSF(ZZ-Z(N))-.000001)4340,4340,4350
4340)   XO#X1(N)
        XD#X2(N)
        ZZ#Z(N+1)
        NN#N+1
        JUMPTO/4360
4350)   XO#X2(N)
        XD#X1(N)
        ZZ#Z(N)
        NN#N-1
4360)   YO#THETA*RADIAN*R(N)
        DY#DTHETA(N)*R(N)*RADIAN
        YD#YO+DY
        DENOM#XD-XO
        SLPE #DY/DENOM
4370)   IF(YD-R2(N))4390,4380,4380
4380)   YI#R2(N)
        XI#(SLPE *XO+YI-YO)/SLPE
        GOTO/(POINT/XO,YO)
        DRAFT/ON
        GOTO/(POINT/XI,YI)
        DRAFT/OFF
        XO#XI
        YO#0
        YD#YD-R2(N)
        JUMPTO/4370
4390)   GOTO/(POINT/XO,YO)
        DRAFT/ON
        GOTO/(POINT/XD,YD)
        DRAFT/OFF
        THETA#YD/(R(N)*RADIAN)
        IF(ABSF(ZZ-Z(J))-.000001)4400,4400,4410
4400)   ZFLAG#ZFLAG+1
        IF(ZFLAG-2*NUMBER+.1)4410,4410,4430
4410)   N#NN
        JUMPTO/4010
4420)   PRINT/0
TITLES  LINE CONNECTING POINTS IS PARALLEL TO L2(N)
        PRINT/3,N,A1,B1,A2,B2,D2,PHI(N),XO,YO,XD,YD,K(N)
4430)   TERMAC
```

REFERENCES

(1) Gold, S. P.; Filament Winding a Surface of Revolution, SCL-DR-65-102; Sandia Corporation Livermore Laboratory, Livermore, California; February 1966.

(2) Bookhart, T. W. and Fowler, A. H.; Computer Programs for Determining Payout Eye Positions for Laying Filaments Along Geodesic Paths on Surfaces of Revolution, Y-1620; Union Carbide Corporation - Nuclear Division Y-12 Plant, Oak Ridge, Tennessee (in preparation).

(3) Lyusternik, L. A.; Shortest Paths Variational Problems, translated by P. Collins and Robert B. Brown; The Macmillan Company, New York, New York (1964).

(4) Bolza, Oskar, Lectures on the Calculus of Variation, 2nd Edition; Chelsea Publishing Company, New York, New York (1960).

(5) Bliss, Gilbert A.; Lectures on the Calculus of Variation; The University of Chicago Press, Chicago, Illinois (1964).

(6) Fowler, A. H. and Wilson, C. W.; Cubic Spline, A Curve Fitting Routine, Y-1400 (Revision 1), pp 104-109; Union Carbide Corporation - Nuclear Division, Y-12 Plant, Oak Ridge, Tennessee; June 28, 1966.